[illegible]
[illegible]
[illegible]

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

S5522

TWO-DIMENSIONAL AUTOREGRESSIVE MODELING

by

Gregory H. Skinner

SEPTEMBER 1989

Thesis Advisor Murali Tummala

Approved for public release; distribution is unlimited.

Unclassified
curity classification of this page

REPORT DOCUMENTATION PAGE

1a Report Security Classification Unclassified		1b Restrictive Markings	
2a Security Classification Authority		3 Distribution Availability of Report	
2b Declassification Downgrading Schedule		Approved for public release; distribution is unlimited.	
4 Performing Organization Report Number(s)		5 Monitoring Organization Report Number(s)	
6a Name of Performing Organization Naval Postgraduate School	6b Office Symbol *(if applicable)* 32	7a Name of Monitoring Organization Naval Postgraduate School	
6c Address *(city, state, and ZIP code)* Monterey, CA 93943-5000		7b Address *(city, state, and ZIP code)* Monterey, CA 93943-5000	
8a Name of Funding Sponsoring Organization	8b Office Symbol *(if applicable)*	9 Procurement Instrument Identification Number	
8c Address *(city, state, and ZIP code)*		10 Source of Funding Numbers	
		Program Element No	Project No / Task No / Work Unit Accession No

11 Title *(include security classification)* TWO-DIMENSIONAL AUTOREGRESSIVE MODELING

12 Personal Author(s) Gregory H. Skinner

13a Type of Report	13b Time Covered	14 Date of Report *(year, month, day)*	15 Page Count
Master's Thesis	From To	SEPTEMBER 1989	70

16 Supplementary Notation The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

17 Cosati Codes			18 Subject Terms *(continue on reverse if necessary and identify by block number)*
Field	Group	Subgroup	two-dimensional sequence, linear prediction, lattice.

19 Abstract *(continue on reverse if necessary and identify by block number)*

Linear prediction has become an important tool for stationary time series analysis. The all-pole system model which is a product of the linear predictive approach has applications in numerous engineering problems. This thesis develops a simple method for obtaining the two-dimensional all-pole system model. The lattice structures that can be used to implement the prediction error and synthesis filters are also shown to have an analogous two-dimensional counterpart. The construction of these filters is in terms of orthogonal Szego polynomials which can be used to solve the two-dimensional block Toeplitz normal equations in a recursive manner. This recursion not only leads to the two-dimensional (2-D) lattice structure, but also allows for expansion of the filter order without resolving the normal equations. Several examples are presented using the two-dimensional linear prediction results for spectral estimation and signal synthesis.

20 Distribution Availability of Abstract	21 Abstract Security Classification	
☒ unclassified unlimited ☐ same as report ☐ DTIC users	Unclassified	
22a Name of Responsible Individual Murali Tummala	22b Telephone *(include Area code)* (408) 646-3214	22c Office Symbol 62Tm

DD FORM 1473,84 MAR 83 APR edition may be used until exhausted
All other editions are obsolete

security classification of this page

Unclassified

T245499

Approved for public release; distribution is unlimited.

Two-Dimensional Auto-Regressive
Modeling

by

Gregory H. Skinner
Lieutenant, United States Navy
B.S.E.E., Auburn University, 1981

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
SEPTEMBER 1989

ABSTRACT

Linear prediction has become an important tool for stationary time series analysis. The all-pole system model which is a product of the linear predictive approach has applications in numerous engineering problems. This thesis develops a simple method for obtaining the two-dimensional all-pole system model. The lattice structures that can be used to implement the prediction error and synthesis filters are also shown to have an analogous two-dimensional counterpart. The construction of these filters is in terms of orthogonal Szego polynomials which can be used to solve the two-dimensional block Toeplitz normal equations in a recursive manner. This recursion not only leads to the two-dimensional (2-D) lattice structure, but also allows for expansion of the filter order without resolving the normal equations. Several examples are presented using the two-dimensional linear prediction results for spectral estimation and signal synthesis.

TABLE OF CONTENTS

I. INTRODUCTION 1
A. LINEAR PREDICTION 1
B. ONE-DIMENSIONAL LINEAR PREDICTION 2
1. All-Pole Model 2
2. Normal Equations and Orthogonality 5
3. Applications of the Autoregressive Model 7
C. NOTATIONAL CONVENTIONS 7
1. Matrix Notation 7
2. Causal Samples 7
3. Polynomial Notation 8

II. TWO-DIMENSIONAL LINEAR PREDICTION 10
A. PROBLEM DEFINITION 10
B. TWO-DIMENSIONAL ORTHOGONALITY 12
1. Description 12
2. Derivation 12
3. Two-Dimensional Normal Equations 14
C. SZEGO POLYNOMIALS 17
1. Levinson Algorithm in Terms of Szego Polynomials 18
2. 2-D Extension 20
D. LATTICE IMPLEMENTATION 24
1. Analysis Lattice 24
2. Synthesis Lattice 28

III. COMPUTER IMPLEMENTATION AND RESULTS 34
A. IMPLEMENTATION 34
1. Autocorrelation Function 34
2. Reflection Coefficients 34
3. Filter Coefficients 35
B. RESULTS 36
1. System Model 36

2. Spectral Estimation 36
3. Signal Synthesis 39

IV. CONCLUSIONS 60

LIST OF REFERENCES 61

INITIAL DISTRIBUTION LIST 62

LIST OF FIGURES

Figure 1. Prediction error and system model filters. 4
Figure 2. Sample space for N1 = N2 = 4. 8
Figure 3. Causal Data Space 9
Figure 4. Generation of the prediction error 11
Figure 5. A visualization of the orthogonality principle for two samples. 13
Figure 6. Variables used to form the bivariate orthogonal polynomials 21
Figure 7. First row of the analysis lattice 25
Figure 8. The lower order polynomials for 26
Figure 9. Coefficient relationship between orthogonal polynomials. 29
Figure 10. One cell of a 2-D analysis lattice 30
Figure 11. First row of the synthesis lattice 32
Figure 12. e_{00} node of the synthesis lattice 33
Figure 13. System models for Eq.136 with an impulse input 37
Figure 14. System models for Eq.136 with a white noise sequence input 38
Figure 15. Generating model for spectral estimates 40
Figure 16. Surface plot of spectral estimate 41
Figure 17. Contour plot of spectral estimate 42
Figure 18. Surface plot of spectral estimate 43
Figure 19. Contour plot of spectral estimate 44
Figure 20. Surface plot of spectral estimate 45
Figure 21. Contour plot of spectral estimate 46
Figure 22. Surface plot of spectral estimate 47
Figure 23. Contour plot of spectral estimate 48
Figure 24. Surface plot of spectral estimate 49
Figure 25. Contour plot of spectral estimate 50
Figure 26. Surface plot of spectral estimate 51
Figure 27. Contour plot of spectral estimate 52
Figure 28. Surface plot of spectral estimate 53
Figure 29. Contour plot of spectral estimate 54
Figure 30. Surface plot of spectral estimate 55
Figure 31. Contour plot of spectral estimate 56

Figure 32. Surface plot of spectral estimate 57
Figure 33. Contour plot of spectral estimate 58
Figure 34. Spectrum of signal generated from white noise 59

I. INTRODUCTION

A. LINEAR PREDICTION

The initial goal of linear prediction is to estimate the present value of a random sequence based on a linear combination of all the past values. This is equivalent to determining a linear, shift invariant, causal prediction error filter which whitens the random sequence. There has been a great deal of study of one-dimensional linear prediction analysis and the usefulness of the resulting filters. The purpose of this thesis is to extend in a straightforward manner the results of one-dimensional linear prediction to two-dimensional signals.

There are several efficient methods for the solution of the normal equations that arise in one-dimensional linear prediction problems. The most familiar of these methods is the Levinson algorithm. The Levinson algorithm alleviates the need to perform a matrix inversion in the solution of the normal equations for the filter coefficients. The recursive nature of the algorithm also allows the prediction error filter to be implemented in a lattice structure. These two features will be of primary concern in the solution of the two dimensional problem.

There are two general approaches to linear prediction. The first method, which is usually referred to as the spectral factorization method, begins with the knowledge of the power spectral density of the random process and attempts to find the model parameters that fit the spectrum. This works well in one-dimensional applications but cannot be extended to two-dimensional signals. In general, a two-dimensional spectrum is not factorable into two polynomials. Although progress has been made in factoring two-dimensional spectra in terms of the complex cepstrum and the results are theoretically exact, these theoretical results cannot be attained in practice. We will, therefore, focus our attention on a second method of attaining the prediction error filter.

The solution method dealt with in this thesis begins with the individual samples of the random process and develops the prediction-error filter from these samples. This method is known as autoregressive model fitting or all-pole modeling. The solution will be developed in terms of orthogonal polynomials that will not only yield a Levinson-like algorithm but are easily extended to two-dimensional signals.

B. ONE-DIMENSIONAL LINEAR PREDICTION

1. All-Pole Model

We are given a finite length sequence of the past values of the random sequence Y(n). The problem is to estimate Y(n) based on a linear combination of these samples. For the simplest case we can estimate Y(n) based solely on the previous sample. The estimate, denoted by $\hat{Y}(n)$, is

$$\hat{Y}(n) = aY(n-1) \tag{1}$$

The prediction will have some error, $e(n)$, which will be minimized in some sense by the choice of a. The least-squares minimization criterion is commonly used. This criterion provides a simple approach to the solution of the linear-prediction problem and thus the modeling problem. We proceed as follows.

First, we redefine the estimate as

$$\hat{Y}(n) = -aY(n-1) \quad . \tag{2}$$

Now the prediction error will be

$$\begin{aligned} e(n) &= Y(n) - \hat{Y}(n) \\ &= Y(n) + aY(n-1) \quad . \end{aligned} \tag{3}$$

The squared error is to be minimum with respect to the coefficient a. This is easily accomplished using derivatives as shown below. Differentiating $[e^2(n)]$ with respect to a gives

$$\frac{d}{da}[e^2(n)] = 2e_n \frac{d}{da} e_n = 0 \tag{4}$$

or

$$[Y(n) + aY(n-1)] \frac{d}{da}[Y(n) + aY(n-1)] = 0 \tag{5}$$

which yields

$$[Y(n) + aY(n-1)][Y(n-1)] = 0 \quad . \tag{6}$$

Finally, solving for a

$$a = -\frac{Y(n)Y(n-1)}{Y(n-1)Y(n-1)} \quad . \tag{7}$$

Since the value of a must minimize the error, $e(n)$, for all values of n, the value of a in (7) could be rewritten as

$$-a = \frac{\sum_{i=1}^{n} Y(i)Y(i-1)}{\sum_{i=1}^{n} Y(i-1)Y(i-1)} \tag{8}$$

Note that (8) will select a such that $\sum_n e^2(n)$ is minimum. It can be seen that the coefficient a is a function of the autocorrelation elements of the given sample space, and this method is usually referred to as the autocorrelation or Yule-Walker approach.

If the order of the predictor is now increased such that

$$\hat{Y}(n) = -\left[a_1 Y(n-1) + a_2 Y(n-2) + \cdots\right] \tag{9}$$

the prediction error becomes

$$e_n = Y(n) - \hat{Y}(n) = Y(n) + a_1 Y(n-1) + a_2 Y(n-2) + \cdots \quad . \tag{10}$$

If (10) is considered to be the difference equation of an all-zero filter, the prediction-error filter can be described by taking the z transform of (10) as

$$E(z) = Y(z)A(z) \tag{11}$$

where

$$A(z) = 1 + a_1 z^{-1} + a_2 z^{-2} + \cdots \quad . \tag{12}$$

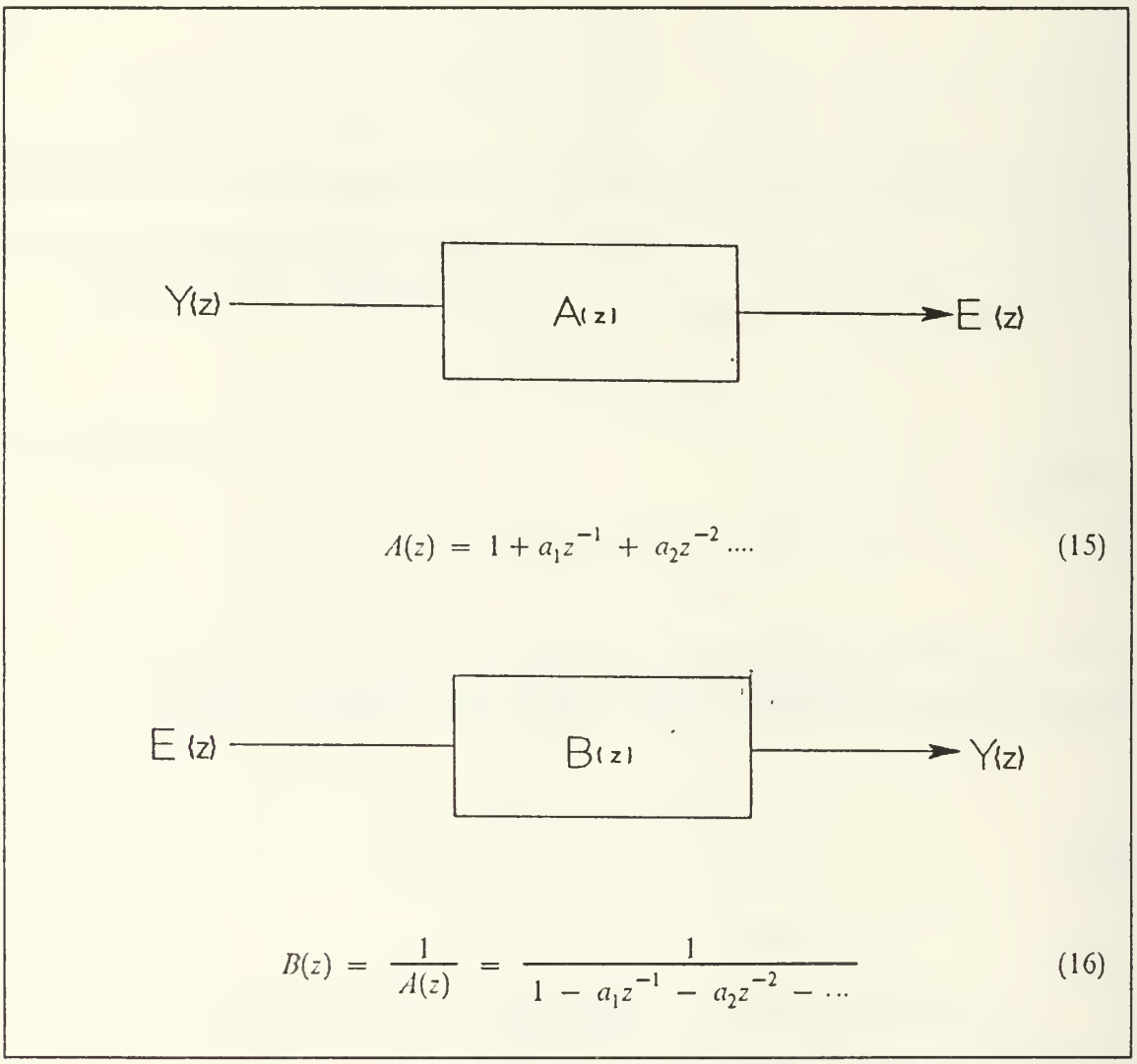
Now if (10) is rewritten and $e(n)$ is taken to be the input to the system, the estimated signal generator model results

$$Y(z) = E(z)B(z) = \frac{1}{A(z)} E(z) \tag{13}$$

where

$$B(z) = \frac{1}{1 + a_1 z^{-1} + a_2 z^{-2} + \cdots} \quad . \tag{14}$$

It can be shown that e_n is a white noise sequence with variance $E[e_n^2]$. This implies that Y(z) and, consequently, Y(n) can be generated from a white noise sequence as shown in Figure 1.



$$A(z) = 1 + a_1 z^{-1} + a_2 z^{-2} \cdots \tag{15}$$

$$B(z) = \frac{1}{A(z)} = \frac{1}{1 - a_1 z^{-1} - a_2 z^{-2} - \cdots} \tag{16}$$

Figure 1. Prediction error and system model filters.

This assumes that the sequence A(z) is causal and causally invertible. If this is the case, we can generate the error sequence from Y(n) and vice-versa by an invertible filter op-

eration [Ref. 1]. The signal generator model, B(z), is the best estimator in a least-squares sense and A(z) is the prediction-error filter. These filters can be used in a number of ways and the simple solution for the filter coefficients is the primary reason for the popularity of linear prediction in signal processing.

2. Normal Equations and Orthogonality

Recalling that we wish to minimize the sum of squares of the error sequence leads directly to an important result in linear prediction known as the orthogonality relationship between the sample space and the error. This principle states that the error $e(n)$ is orthogonal to all the samples of Y(n) used in obtaining the estimate $\hat{Y}(n)$. That is,

$$E[e(n), Y(n-i)] = 0, \quad 1 \le i \le N \tag{17}$$

where N is the order of the predictor. This result is derived for the two-dimensional case in Chapter II. It is the central idea behind the solution of the linear prediction problem in terms of orthogonal polynomials. It also leads directly to the normal equations in both one- and two-dimensional formulations.

The derivation of the normal equations is a simple extension of the orthogonality relationship (17). Suppose we seek a kth-order predictor and the value of the prediction-error variance. We will require (k+1) linearly independent equations to solve for the k prediction coefficients and the prediction-error variance. These coefficients can be derived as follows.

$$\begin{aligned} e_n &= Y(n) - \hat{Y}(n) \\ &= Y(n) + a_1 Y(n-1) + a_2 Y(n-2) + \cdots + a_k Y(n-k) \end{aligned} \tag{18}$$

Substituting (18) into (17) gives

$$E[Y(n) + a_1 Y(n-1) + a_2 Y(n-2) + \cdots, Y(n-i)] = 0 \quad 1 \le i \le N \tag{19}$$

If we now define the coefficient $a_0 = 1$ and note that for a stationary process the autocorrelation, $R(x)$, is given by $E[Y(k)Y(l)] = R_{k-l}$, (19) may now be written as

$$E\left[\sum_{k=0}^{N} a_k Y(n-k), Y(n-i)\right] = 0 \tag{20}$$

$$\sum_{k=0}^{N} a_k R(i-k) = 0 \quad 1 \le i \le N \tag{21}$$

where N is the order of the predictor being used and R is the autocorrelation function of the random sequence Y(n). The N equations generated by (21) can be written in matrix form for N = 2 as

$$\begin{bmatrix} R_{-1} & R_0 & R_{-1} \\ R_2 & R_1 & R_0 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} . \tag{22}$$

The error variance ε is given by

$$\begin{aligned} \varepsilon &= E\left[e_n^2\right] = E\left[e_n, Y(n) - \hat{Y}(n)\right] \\ &= E[e_n, Y(n)] . \end{aligned} \tag{23}$$

Substituting (18) into (23) yields

$$\varepsilon = E\left[\sum_{k=0}^{N} a_k Y(n-k), Y(n)\right] \tag{24}$$

which in terms of the autocorrelation becomes $\varepsilon = \sum_{k=0}^{N} a_k R_{-k}$. For example, with N = 2, this can be combined with (22) to yield

$$\begin{bmatrix} R_0 & R_{-1} & R_{-2} \\ R_1 & R_0 & R_{-1} \\ R_2 & R_1 & R_0 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} \varepsilon \\ 0 \\ 0 \end{bmatrix} \tag{25}$$

This autocorrelation matrix is seen to be (N + 1) by (N + 1), Toeplitz, and since $R_{-k} = R_k$, it is symmetric. The various solutions for (25) occupy much of the literature on linear prediction. For our purposes, the desired solution must provide a simple extension to two-dimensional signals and be adaptable to a lattice structure. J.H. Justice [Ref. 2] has derived such a solution in terms of Szego polynomials. Although quite similar to the Levinson algorithm, it has the important advantage of not relying on the symmetry of the correlation matrix. The two-dimensional autocorrelation matrix will be found to be Toeplitz in blocks, and, although this provides some simplification, the

symmetry of (25) is lost. Fortunately, a solution in terms of orthogonal Szego polynomials is easily derived [Ref. 3].

3. Applications of the Autoregressive Model

Linear prediction plays an integral role in the solution of a number of engineering problems. The resulting model can be used for high resolution spectral estimation [Ref. 4]. Speech signals can be reduced from a large volume of data to a comparatively small data space by linear predictive coding [Ref. 5]. A model of the generating function for the random sequence can be derived. Lattice filters can be implemented which can be used to form the prediction-error filter in the forward direction or synthesize the random process in the inverse direction.

The many applications of linear prediction in time series analysis has generated a great deal of interest in extending the results to multi-dimensional random processes. Multi-dimensional signals occur in situations where the samples are spatially dependent such as radar, sonar, and image processing. The signals considered in this thesis are wide sense stationary and scalar-valued sequences of two variables. The two-dimensional autoregressive model that is derived will be used in some of the same applications classically associated with the 1-D model including spectral estimation and sequence generator estimation. An alternative implementation of the model in a lattice structure will also be presented.

C. NOTATIONAL CONVENTIONS

1. Matrix Notation

This thesis will deal with a finite size rectangular sample space. The individual samples will be denoted by $Y(n_1 - k, n_2 - l)$, where k and l are measures of distance from the point at which the value of the sequence is being estimated. The sample space will contain N1 x N2-1 samples. Figure 2 shows an example of the sample space with $N1 = N2 = 4$. In this case we would be predicting the value of Y(3,3) based on the 15 samples available. Notice that, in this case, k and l have a maximum value of 3. Spatial samples are indexed by n1 and n2 to denote rows and columns, or the vertical and horizontal directions, respectively.

2. Causal Samples

In one-dimensional signals the clear meaning of past and future leads to a straightforward definition of causality. In two-dimensional signals the ordering of the points is of central importance; therefore a reference for past and future needs to be established. In this thesis causal samples will be taken to be in the quarter plane shown

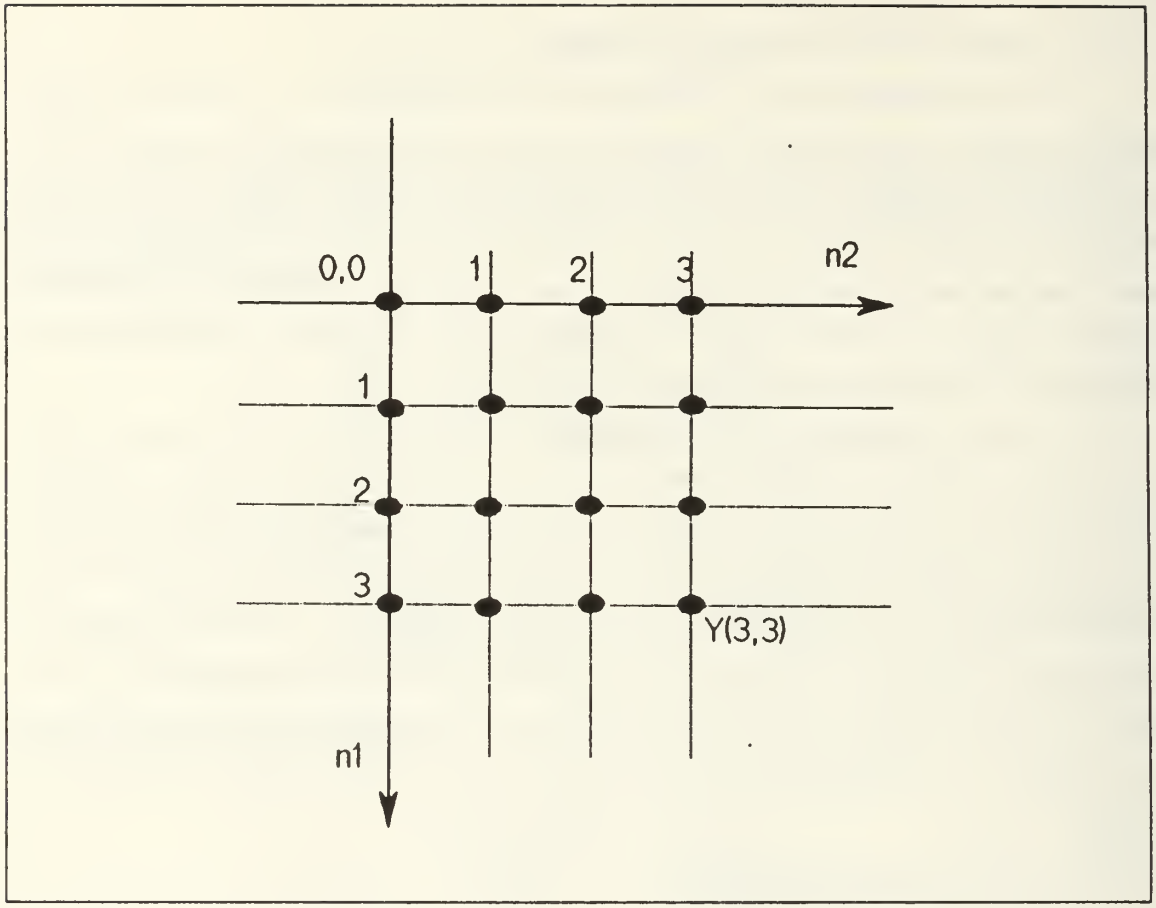


Figure 2. Sample space for N1 = N2 = 4.

in Figure 3. This is equivalent to describing the desired filter as a third-quadrant filter. Due to the symmetry of the 2-D autocorrelation the third and first quadrants produce the same results. However, causal will be used only in reference to the samples shown in Figure 3.

3. Polynomial Notation

Two-dimensional polynomials will be denoted by $\mathbf{P}_{kl}$ where k is the degree of the polynomial in the n1 direction and l is the degree in the n2 direction. When the need to refer to an individual element in a polynomial arises, the element will be identified as $p_{kl}(i,j)$, where i and j give the power of the bivariate variable. The correspondence between the z transform of the system model and the coefficients in the orthogonal polynomials leads to the use of z_1 and z_2 as the independent variables in this study. Thus the polynomial $\mathbf{P}_{22}$ will be a 3x3 bivariate polynomial consisting of the elements

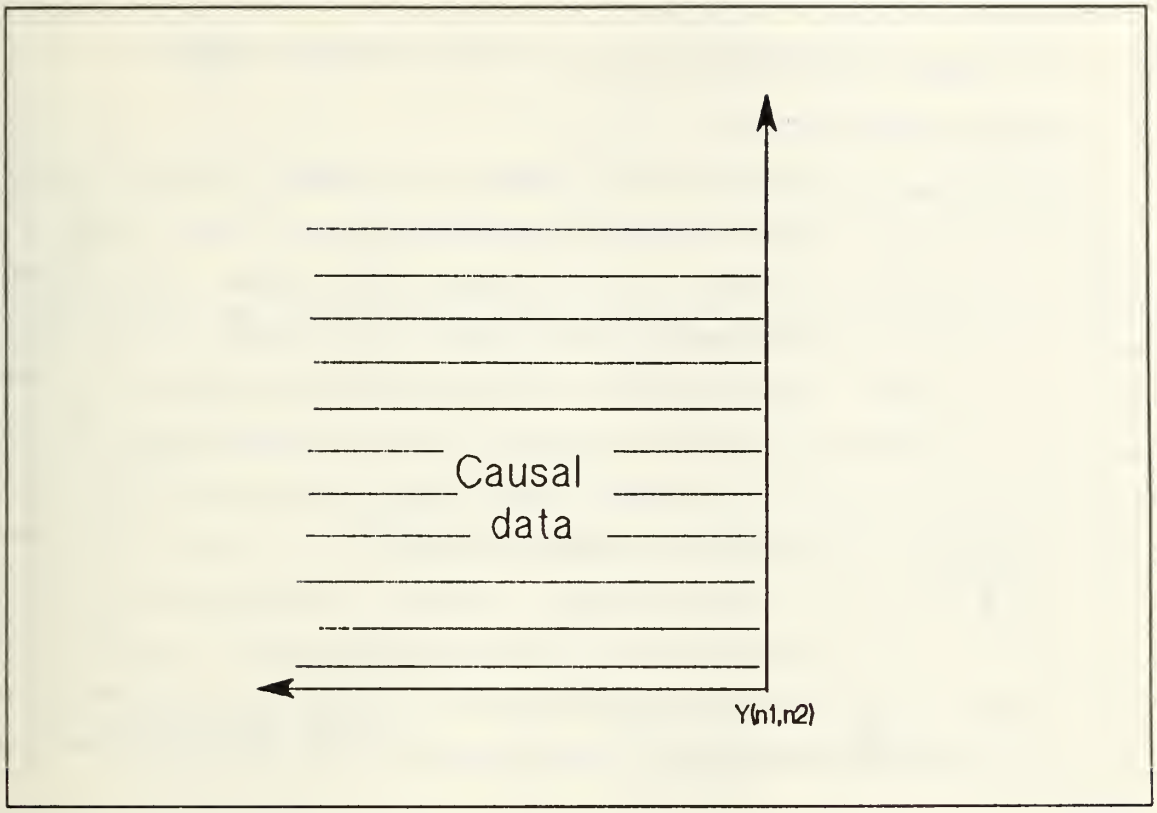


Figure 3. Causal Data Space

$$\mathbf{P}_{22} = \sum \begin{matrix} p_{22}(0,0) & p_{22}(0,1)z_2 & p_{22}(0,2)z_2^2 \\ p_{22}(1,0)z_1 & p_{22}(1,1)z_1z_2 & p_{22}(1,2)z_1z_2^2 \\ p_{22}(2,0)z_1^2 & p_{22}(2,1)z_1^2z_2 & p_{22}(2,2)z_1^2z_2^2 \end{matrix} \tag{26}$$

The development of $\mathbf{P}_{kl}$ is in terms of positive powers of z. The desired filter must be causal and therefore in negative powers of z. This is easily accomplished by multiplying $\mathbf{P}_{kl}$ by $z_1^{-k}z_2^{-l}$. The resulting causal filter will be denoted $\mathbf{A}_{kl}$. The leading coefficient of $\mathbf{P}_{kl}$ is $p_{kl}(k,l)$ which will become $a_{kl}(0,0)$ in the causal filter.

II. TWO-DIMENSIONAL LINEAR PREDICTION

A. PROBLEM DEFINITION

The two-dimensional linear prediction problem is conceptually identical to the one-dimensional problem. That is, we attempt to predict the value of Y(n1,n2) based on a linear combination of the previous N1xN2 rectangular block of data. The coefficients which result in the optimum prediction will comprise a linear, shift invariant, causal, prediction error filter. The problem is to find the coefficients that minimize in some sense the error between the actual value of Y(n1,n2) and the predicted value, $\hat{Y}$(n1,n2). This will require the solution of a block-Toeplitz matrix which contains N1xN1 blocks and each block will contain N2xN2 correlation coefficients. We will require the solution to provide the means for a direct realization of the filter or implementation in a lattice structure. The mean square error criterion is again utilized to obtain the optimum prediction filter. If a solution to the desired filter can be achieved recursively, then the relationship between successively higher order filters can be used to implement the lattice filter.

We begin by seeking a solution to the linear prediction problem, and will assume that Y(n1,n2) is a two-dimensional stationary random sequence. Given the causal block of N1xN2 data samples we need to generate $\hat{Y}$(n1,n2) from a linear combination of the data that minimizes the prediction error. The prediction, $\hat{Y}$(n1,n2), can be written as

$$\begin{aligned}\hat{Y}(n1,n2) = & -a_{10}Y(n1-1,n2) - a_{01}(n1,n2-1) - a_{11}Y(n1-1,n2-1) - \cdots \\ & \cdots - a_{N1-1,N2-1}Y(n1-N1+1,n2-N2+1)\end{aligned} \tag{27}$$

where the a_{st}'s are the filter coefficients associated with data samples that are delayed by s in the n1 direction and t in the n2 direction. The coefficients can be assembled in a matrix to yield the desired two-dimensional linear prediction filter. This filter will be an all-zero model with the finite sample space Y as the input and the prediction, $\hat{Y}$, as the output as described below.

$$\hat{Y} = \begin{bmatrix} a_{N1-1,N2-1} & \cdots & a_{N1-1,1} & a_{N1-1,0} \\ \vdots & & \vdots & \vdots \\ a_{1,N2-1} & \cdots & a_{11} & a_{10} \\ a_{0,N2-1} & \cdots & a_{01} & 0 \end{bmatrix} Y \quad \text{2-D filter coefficients} \qquad (28)$$

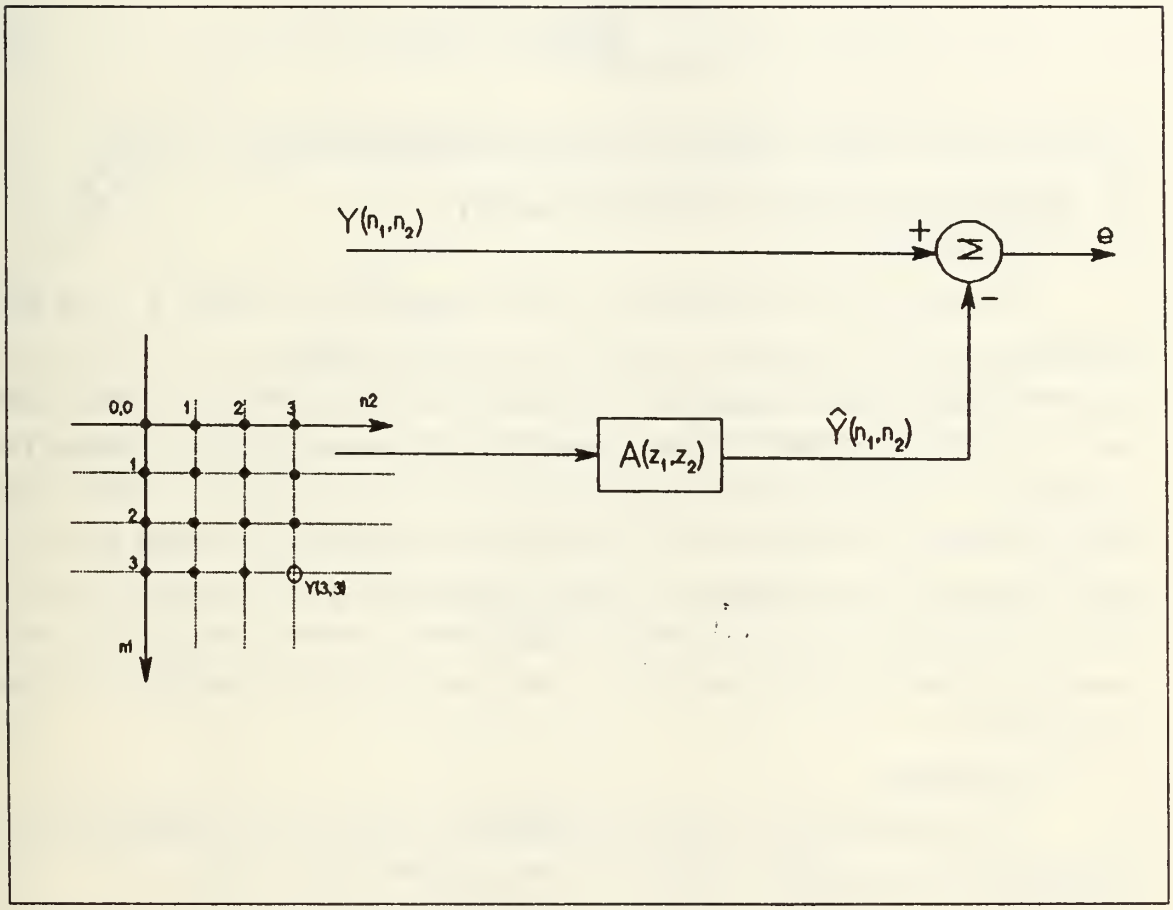


Figure 4. Generation of the prediction error

The prediction process can be depicted as in Figure 4. The polynomial $A(z_1, z_2)$ is simply the z transform of (27) given by

$$A(z_1, z_2) = -(a_{10}z_1^{-1} + a_{01}z_2^{-1} + a_{11}z_1^{-1}z_2^{-2} + \cdots) \ . \qquad (29)$$

For any choice of the coefficients some error will be present. This prediction error is as seen in Figure 4 and can be written as

$$e = Y(n1,n2) + \sum_{s=0}^{N1-1} \sum_{t=0}^{N2-1} a_{st} Y(n1-s,n2-t) \ . \ (s,t) \neq (0,0) \tag{30}$$

We will define the filter coefficient $a_{00} = 1$ and condense (30) to the form

$$e = \sum_{s=0}^{N1-1} \sum_{t=0}^{N2-1} a_{st} Y(n1-s,n2-t) \ . \tag{31}$$

The task now is to find the filter coefficients that minimize the error.

B. TWO-DIMENSIONAL ORTHOGONALITY

1. Description

The error, Y(n1,n2)-$\hat{Y}(n1,n2)$, can be visualized as in Figure 5. Any linear combination of Y(n1-1,n2) and Y(n1,n2-1) will yield an estimate of $\hat{Y}$ which lies in the plane of Y(n1-1,n2) and Y(n1,n2-1). If Y(n1,n2) lies in this plane, then $\hat{Y}$ will be exact and there will be no error. For any other Y(n1,n2) the error will be minimum if it is orthogonal to the sample space spanned by Y(n1-1,n2) and Y(n1,n2-1). This is equivalent to defining $\hat{Y}$ as that portion of Y(n1,n2) that is parallel to the sample space. Thus Y-$\hat{Y}$ leaves only the component of Y that is perpendicular to the sample space as the error. Although it can not be drawn for higher order predictors, the minimum error is always such that it is orthogonal to the space spanned by all the samples of Y used in the estimate.

2. Derivation

The orthogonality of the error to the sample space can be easily derived. First we define the inner product, $<X,Y>$, of two sequences, X and Y, by

$$<X,Y> = \sum_k \sum_l X_{kl} Y^*_{kl} . \tag{32}$$

Now the inner product of the squared error can be expressed as

$$<e^2> = <(Y - \hat{Y})^2> \tag{33}$$

and we seek a minimum squared error with respect to the filter coefficients $\vec{a}$.

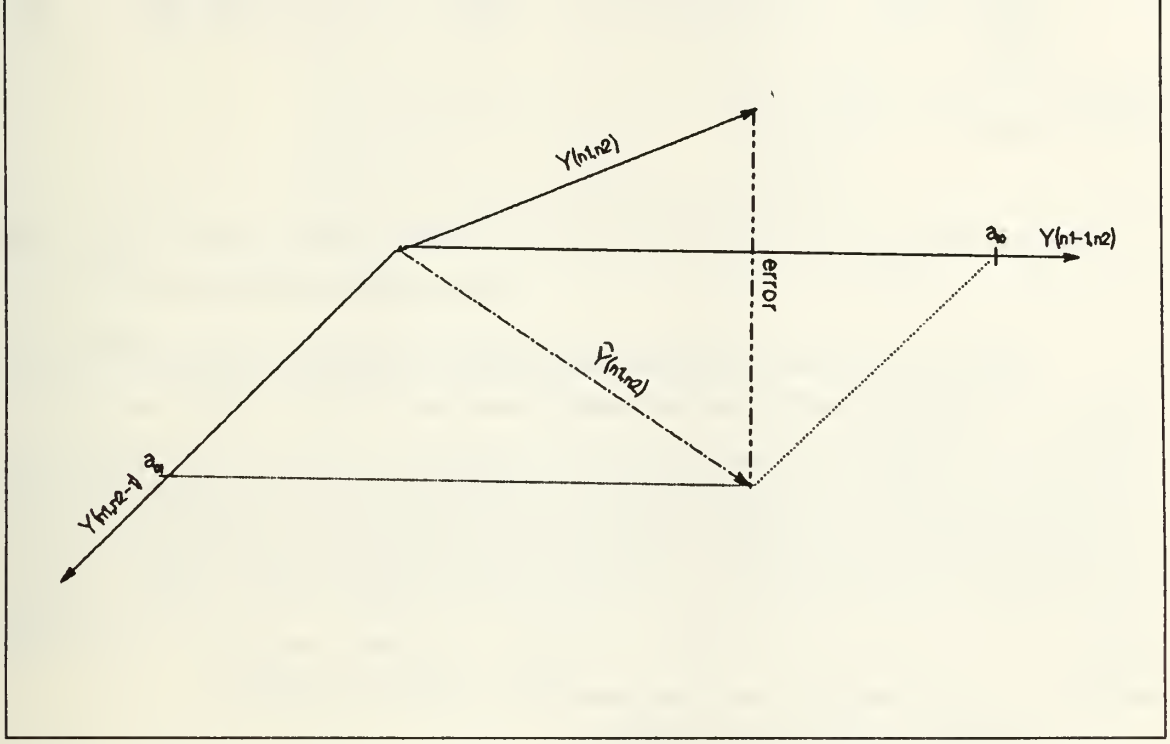


Figure 5. A visualization of the orthogonality principle for two samples.

$$\frac{d}{d\vec{a}} < e^2 > = 0 \tag{34}$$

or

$$< 2e \frac{d}{d\vec{a}} e > = 0 \tag{35}$$

where $\frac{d}{d\vec{a}}$ are the partial derivatives with respect to the elements in the coefficient matrix. Substituting (30) for e

$$\frac{d}{d\vec{a}} e = \frac{d}{d\vec{a}} [Y(n1,n2) + \sum_{s=0}^{N1-1} \sum_{t=0}^{N2-1} a_{st} Y(n1 - s, n2 - t)] \tag{36}$$

leads to

$$\frac{d}{d\vec{a}}\, e = [\,Y(n1-1,n2), Y(n1,n2-1), \cdots\,, Y(n1-N1-1,n2-N2-1)]\ . \qquad (37)$$

Substituting (37) into (35) yields

$$\begin{aligned} < e, Y(n1,-s,n2-t) > \ &= 0\ , \quad 0 \le s < N1 \\ &\qquad\quad 0 \le t < N2 \\ &\qquad\quad s,t \neq 0\ . \end{aligned} \qquad (38)$$

This result, (38), states that the inner product of the error and each of the past samples of Y(n1,n2) used in the estimate is zero which requires that the error be orthogonal to the sample space. Not only is the error orthogonal to the sample space, but errors created by successively higher-order predictors are orthogonal to each other. Also, the sequence $e(n_1, n_2)$ will be uncorrelated to the degree that the order of the predictor allows. That is, if we had access to an infinite causal data space, then we would produce a purely white error sequence. Since we will always be limited to some finite size filter, we will produce an error sequence that is partially correlated.

3. Two-Dimensional Normal Equations

The sequences considered in this thesis are taken to be wide-sense stationary. If they are also taken to be zero-mean signals, the inner product of two sequences is equivalent to the expected value of the product of the sequences. The expected value of Y(n1,n2) and Y(n1-s,n2-t) reduces to a two-dimensional autocorrelation that is solely a function of the delays s and t. Thus,

$$< Y(n1,n2), Y(n1-s,n2-t) > \ = \ E[\,Y(n1,n2), Y(n1-s,n2-t)] \ = \ R_{st} \qquad (39)$$

where R will be used to denote the autocorrelation with lags s in the n1 direction and t in the n2 direction. Equation (38) can now be used to form a block-Toeplitz matrix that is analogous to the Toeplitz matrix generated by the normal equations in one-dimensional linear prediction. Thus, we can write

$$E[e,\ Y(n1-s,n2-t)] = 0 \quad (s,t) \neq (0,0) \qquad (40)$$

substituting for e

$$E\left[\, Y(n1,n2) + \sum_{k=0}^{N1-1} \sum_{l=0}^{N2-1} a_{kl} Y(n1-k,n2-l)\ ,\ Y(n1-s,n2-t) \right] = 0 \quad \begin{matrix} 0 \le s < N1 \\ 0 \le t < N2 \\ (s,t) \neq (0,0) \end{matrix} \qquad (41)$$

that can be written as

$$\sum_{k=0}^{N1-1}\sum_{l=0}^{N2-1} a_{kl}R(s-k,t-l) = 0 \quad \begin{matrix} 0 \le s < N1 \\ . \ 0 \le t < N2 \\ s,t \neq 0 \end{matrix} \tag{42}$$

For example, with a N1 = N2 = 2, and with the first filter coefficient defined as $a_{00} = 1$ the normal equations become

$$E\big[\, Y(n1,n2) + a_{01}Y(n1,n2-1) + a_{10}Y(n1-1,n2) + \\ + a_{11}Y(n1-1,n2-1)\,,\, Y(n1-s,n2-t)\big] = 0 \tag{43}$$

where (42) or (43) are the two dimensional normal equations. They are a direct result of the orthogonality relationship described by (40). These equations can be written out to yield N1xN2-1 linearly independent equations. For example, with N1 = N2 = 2, we have

$$s=0,t=1 \quad R_{01} + a_{01}R_{00} + a_{10}R_{-1,1} + a_{11}R_{-1,0} = 0 \tag{44}$$

$$s=1,t=0 \quad R_{10} + a_{01}R_{1,-1} + a_{10}R_{00} + a_{11}R_{0,-1} = 0 \tag{45}$$

$$s=1,t=1 \quad R_{11} + a_{01}R_{10} + a_{10}R_{01} + a_{11}R_{00} = 0 \tag{46}$$

Equations (44)-(46) can be augmented by the expression for the squared error ε which is given by

$$\varepsilon = E[e^2] = E[e,\, Y - \hat{Y}] \tag{47}$$

or

$$= E\left[\sum_{k=0}^{N1-1}\sum_{l=0}^{N2-1} a_{kl}Y(n1-k,n2-l),Y(n1,n2)\right] \tag{48}$$

and in terms of the autocorellation

$$= \sum_{k=0}^{N1-1}\sum_{l=0}^{N2-1} a_{kl}R(-k,-l)\ . \tag{49}$$

Assembling the above equations into a matrix yields

$$\begin{bmatrix} R_{00} & R_{0,-1} & R_{-1,0} & R_{-1,-1} \\ R_{01} & R_{00} & R_{-1,1} & R_{-1,0} \\ R_{10} & R_{1,-1} & R_{00} & R_{0,-1} \\ R_{11} & R_{10} & R_{01} & R_{00} \end{bmatrix} \begin{bmatrix} a_{00} \\ a_{01} \\ a_{10} \\ a_{11} \end{bmatrix} = \begin{bmatrix} \varepsilon \\ 0 \\ 0 \\ 0 \end{bmatrix} . \tag{50}$$

The resulting matrix is (N1N2)x(N1N2) square and will consist of blocks which are Toeplitz. There will be N1xN1 blocks and each block will contain N2xN2 correlation coefficients. The matrix can be seen to be Toeplitz with respect to the blocks and can be written as

$$\begin{bmatrix} \Phi_0 & \Phi_1^H \\ \Phi_1 & \Phi_0 \end{bmatrix} \begin{bmatrix} \vec{a}_0 \\ \vec{a}_1 \end{bmatrix} = \begin{bmatrix} \vec{\varepsilon} \\ 0 \end{bmatrix} \tag{51}$$

where the Φ_k's are blocks in the autocorrelation matrix and H indicates hermitian transposition. Each Φ_k is an N2xN2 Toeplitz matrix. For the example from (50), we have

$$\Phi_0 = \begin{bmatrix} R_{00} & R_{0,-1} \\ R_{01} & R_{00} \end{bmatrix} \tag{52}$$

$$\Phi_1 = \begin{bmatrix} R_{10} & R_{1,-1} \\ R_{11} & R_{10} \end{bmatrix} \tag{53}$$

$$\vec{a}_0 = (a_{00,}\ a_{01})^T \tag{54}$$

$$\vec{a}_1 = (a_{10,}\ a_{11})^T \tag{55}$$

$$\vec{\varepsilon} = (\varepsilon,\ 0)^T \tag{56}$$

If an element in Φ_k is R_{st}, the corresponding element in Φ^H is $R^*_{-s,t}$. Since we are dealing only with real data the conjugate notation can be dropped from R. It is also worthwhile to note that $R_{st} = R_{-s,-t}$.

The solution of this matrix for the coefficients a_{st} will yield the two-dimensional whitening filter. There has been a great deal of research involving block Toeplitz matrices and there are several algorithms that could be used to obtain the coefficients. In order to implement the filter in a lattice structure, a recursive solution beginning with a zeroth-order filter and proceeding to the desired order is necessary. In one-dimensional

prediction error filtering, the Levinson algorithm successively represents the higher order filters in terms of reflection coefficients. It will be seen that an analogous solution to the two-dimensional prediction error filter can be developed in terms of Szego polynomials.

C. SZEGO POLYNOMIALS

The solution to the system of equations $\Phi\vec{a} = \varepsilon$ does not require that the inverse of the matrix Φ be found. All that is required is an orthogonal set of polynomials that span the space defined by Φ. This result follows the orthogonality relationship of the prediction error and the sample space. For example, if we had a second order and a first order prediction error filter the corresponding prediction errors would be

$$e_1 = Y(1) - a_1(1)Y(0) \tag{57}$$

$$e_2 = Y(2) - a_2(1)Y(1) - a_2(2)Y(0) \tag{58}$$

$$E[e_2, e_1] = E[e_2, Y(1) - a_1(1)Y(0)] \tag{59}$$

But (44) requires that e_2 is orthogonal to both Y(1) and Y(0), and therefore

$$E[e_2, e_1] = 0 \quad . \tag{60}$$

It is evident from the above observations that the choice of $\mathbf{A}_{k-1}$ must be such that it is orthogonal to $\mathbf{A}_k$. If we begin by defining $\mathbf{A}_0$ as any convenient value and proceed to find the successively higher-order orthogonal polynomials until the desired predictor order is reached, the final polynomial will hold the coefficients which will provide the optimum prediction-error filter. To accomplish this we will use Szego polynomials. Szego polynomials are a special class of polynomials which have zero inner products over an interval on the unit circle [Ref. 6]. This is precisely the requirement of the orthogonal polynomials relating successively higher order-prediction errors. J.H. Justice [Ref. 3] has presented a recursive method of generating Szego polynomials of the form

$$P_k = p_{(k)}z^k + p_{k-1}z^{k-1} + \cdots + p_0 \tag{61}$$

where

$$z = e^{jk\theta} \tag{62}$$

The matrix Φ is an inner product matrix which may be utilized to orthonormalize a sequence of polynomials on the unit circle [Ref. 6]. The resulting orthonormal sequence

will be composed of Szego polynomials. If the filter sequence $\mathbf{A}_k$ is determined in terms of Szego polynomials that are orthogonal over the space associated with the autocorrelation matrix , the prediction error filter will result [Ref. 2]. Further, the Szego polynomials can be determined in a recursive manner which is necessary for the representation of the filter in a lattice structure. In a one-dimensional filter this will result in the Levinson algorithm. J.H. Justice [Ref. 2] has developed the bivariate extension of Szego polynomials that can be utilized to orthonormalize the block Toeplitz matrix generated by the two-dimensional autocorrelation of the sequence Y(n1,n2).

1. Levinson Algorithm in Terms of Szego Polynomials

Any positive definite matrix can be defined as an inner product space and can thus be used to define orthogonal polynomials spanning that space. The autocorrelation matrix produces such a space. If the autocorrelation matrix is generated from a sequence, $Y = Y_0, Y_1, \cdots, Y_N$, then the sequence of polynomials, $1, z, z^2, \cdots, z^N$ can be orthonormalized with respect to the autocorrelation matrix. The resulting polynomials are $P_0(z), P_1(z), \cdots, P_N(z)$. The subscript on P denotes the degree of the polynomial. The requirement on $\mathbf{P}_k$ is that it is orthogonal to all polynomials of lesser degree. Thus, we have

$$< P_k, P_l > \; = \; \delta(k - l) \tag{63}$$

A one-dimensional sequence of orthogonal polynomials will consist of the terms

$$\begin{aligned} P_0 &= p_{00} \\ P_1 &= p_{11}z \; + \; p_{10} \\ &\cdot \\ &\cdot \\ &\cdot \\ P_N \; &= \; p_{NN}z^N \; + \; p_{NN-1}z^{N-1} \; + \cdots + \; p_{N0} \end{aligned} \tag{64}$$

and in general,

$$P_k \; = \; \sum_{j=0}^{k} p_{kj} z^j \quad . \tag{65}$$

Justice's method for determining these polynomials recursively results in an algorithm equivalent to the Levinson algorithm. This algorithm for the 1-D case is briefly summarized as follows. We begin by setting

$$P_0 = R_0^{-\frac{1}{2}} \quad . \tag{66}$$

Then the subsequent polynomials can be determined by the recursive relationship

$$P_{n+1}(z) = \left(1 - |\lambda_n|^2 p_{nn}^2\right)^{-\frac{1}{2}} \left(z P_n(z) - \lambda_n p_{nn} P_n^r(z)\right) \tag{67}$$

where

$$P_n^r(z) = z^n P_n(z^{-1}) \tag{68}$$

and

$$\lambda_n = \sum_{k=0}^{n} R_{k+1} p_{nk} \quad . \tag{69}$$

The equation (67) can be expressed in the form of a causal prediction-error filter that yields the familiar form of the Levinson algorithm. The nth-order polynomial produced by (67) will be of the form:

$$P_n(z) = p_{nn} z^n + p_{nn-1} z^{n-1} + \cdots + p_{n0} \quad . \tag{70}$$

The causal prediction-error filter is of the form:

$$A_n(z) = a_{n0} + a_{n1} z^{-1} + \cdots + a_{nn} z^{-n} \tag{71}$$

where we let $a_{np} = p_{n,n-p}$ and $A_n = z^{-n} P_n$. Then multiplying (67) by $z^{-(n+1)}$ yields

$$z^{-(n+1)} P_{n+1}(z) = z^{-n} P_n(z) - z^{-(n+1)} \lambda_n P_n^r(z) \tag{72}$$

where the normalizing factor, $(1 - |\lambda_n|^2 p_{nn}^2)^{-\frac{1}{2}}$, has been absorbed into the definition of λ_n and the leading coefficients are defined as $p_{nn} = a_{n0} = 1$. Now, we define the backwards polynomials as follows

$$P_n^r(z) = z^n P_n(z^{-1}) \tag{73}$$

$$\begin{aligned} A_n^r(z) &= z^{-n} A_n(z^{-1}) \\ &= z^{-n}(z^n P_n(z^{-1})) \end{aligned} \tag{74}$$

$$P_n^r(z) = z^n A_n^r(z) \tag{75}$$

where (72) can now be written as

$$A_{n+1}(z) = A_n(z) - z^{-1}\lambda_n A_n^r(z) \tag{76}$$

$$\begin{aligned} A_{n+1}^r(z) &= z^{-(n+1)}\left(A_n(z^{-1}) - z\lambda_n A_n^r(z^{-1})\right) \\ &= z^{-1}A_n^r(z) - z^{-n}\lambda_n(z^n A_n(z)) \end{aligned} \tag{77}$$

$$A_n^r(z) = z^{-1}A_n^r(z) - \lambda_n A_n(z) \tag{78}$$

$$\lambda_n = \frac{\sum_{i=0}^{n} a_{ni} R_{(N+1-i)}}{\sum_{i=0}^{n} a_{ni} R_{(i)}} \quad . \tag{79}$$

Equations (76), (78) and (79) represent the usual Levinson recursion of one-dimensional linear prediction problems. The coefficients found using the Levinson algorithm will be developed in negative powers of z. The coefficients found using (67) are in positive z and differ by a normalizing factor. Both sets of polynomials, **A** and **P** , will be orthogonal sets of polynomials on the unit circle spanning the same inner product space. Since the definition of the λ_n is simpler using (67), we will proceed to the two dimensional extension using positive powers of z and make the necessary conversion to a causal filter after the coefficients have been found.

2. 2-D Extension

Now we want to extend these ideas to two-dimensional filters. The necessary extension to bivariate orthogonal polynomials has been developed by Justice [Ref. 2]. The derived polynomials will be orthogonal in two variables with respect to the inner product space of Φ as described by

$$E[P_{kl}, P_{st}] = \delta(k-s)\delta(l-t) \quad . \tag{80}$$

The set of polynomials to be orthonormalized are shown in Figure 6. The procedure starts with the first row which is a function of z_2 alone. This row, therefore, can be orthonormalized by the same method presented for one-dimensional systems.

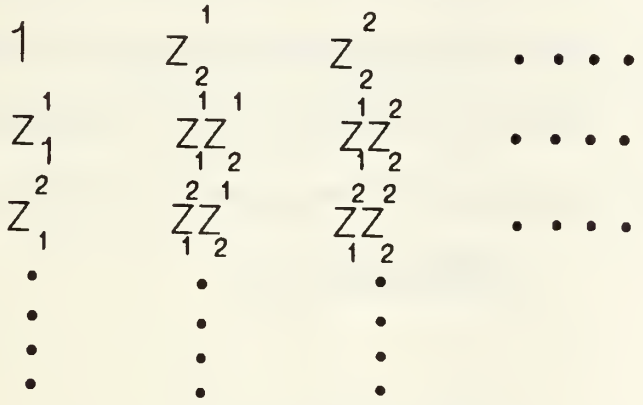


Figure 6. Variables used to form the bivariate orthogonal polynomials

This is equivalent to taking the block Φ_0 from the two-dimensional autocorrelation matrix and finding the filter coefficients $a_{01}, a_{02}, \cdots, a_{0m}$. Thus,

$$\Phi_0 = \begin{bmatrix} R_{00} & R_{0,-1} & \cdots & R_{0,-m} \\ R_{01} & R_{00} & \cdots & R_{0,-(m-1)} \\ & & \ddots & \\ R_{m0} & & \cdots & R_{00} \end{bmatrix} \tag{81}$$

where

$$\vec{a}_0 = (a_{00}, a_{01}, \cdots, a_{0m})^T \quad . \tag{82}$$

It is easily shown that $R_{0m} = R_{0,-m}$, so that this is a symmetric matrix and a recursive solution has been presented. Once the first row of orthogonal polynomials has been determined, the polynomials of the next row can be expressed as a linear combination of the lower-order polynomials. For example, with N2 = 3 the first row will have polynomials P_{00}, P_{01}, and P_{02}. The first polynomial in the second row to be found is P_{10}. This can be accomplished as follows:

First note that if P_{00} is multiplied by z_1 its order is changed and it is no longer a member of the orthogonal set. We may then write

$$z_1 P_{00} = \lambda_{00} P_{10} + \lambda_{01} P_{01} + \lambda_{02} P_{02} \tag{83}$$

and after multiplying on both sides by P_{10} and taking the expected value we find

$$E[z_1 P_{00}, P_{10}] = \lambda_{00} E[P_{10}, P_{10}] + \lambda_{01} E[P_{01}, P_{10}] + \lambda_{02} E[P_{02}, P_{10}] \ . \tag{84}$$

Since P_{10} is orthogonal to all polynomials of lower order in z_1, this reduces to

$$E[z_1 P_{00}, P_{10}] = \lambda_{00} \tag{85}$$

and similarly

$$E[z_1 P_{00}, P_{01}] = \lambda_{01} \tag{86}$$

$$E[z_1 P_{00}, P_{02}] = \lambda_{02} \ . \tag{87}$$

Thus, in general we find that

$$z_1 P_{kl} = \lambda_{kl} P_{k+1,l} + \sum_{t=0}^{l-1} \lambda_{kt} P_{k+1,t} + \sum_{s=0}^{k} \sum_{t=0}^{N2-1} \lambda_{st} P_{st} \, . \tag{88}$$

Now we define the reverse polynomial

$$P^r_{ij}(z_1, z_2) = z_1^k z_2^m P_{ij}(z_1^{-1}, z_2^{-1}) \, . \, 0 \le i \le k \quad 0 \le j \le m \quad m = N2 - 1 \tag{89}$$

Note that P^r_{ij} will also be an orthogonal set of polynomials and will span the same space as P_{ij}. Now (88) may be written in terms of the reverse polynomials

$$z_1 P_{kl} = \lambda_{kl} P_{k+1,l} + \sum_{t=0}^{l-1} \lambda_{kt} P_{k+1,t} + \sum_{s=0}^{k} \sum_{t=0}^{m} \lambda'_{st} P^r_{st} \tag{90}$$

where

$$\lambda'_{st} = E[z_1 P_{kl}, P^r_{st}] \ . \tag{91}$$

But for $0 \le s \le k - 1$

$$E[z_1 P_{kl}, P^r_{st}] = 0 \tag{92}$$

so that (90) may be reduced to

$$z_1 P_{kl} = \lambda_{kl} P_{k+1,l} + \sum_{t=0}^{l-1} \lambda_{kt} P_{k+1,t} + \sum_{t=0}^{m} \lambda'_{kt} P^r_{kt} \tag{93}$$

where

$$\lambda_{kt} = E[z_1 P_{kl}, P_{k+1,t}] \tag{94}$$

and

$$\lambda'_{kt} = E[z_1 P_{kl}, P^r_{kt}] \quad . \tag{95}$$

The polynomials, $P_{k+1,l}$, may now be found from the lower order polynomials

$$\lambda_{kl} P_{k+1,l} = z_1 P_{kl} - \sum_{t=0}^{l-1} \lambda_{kt} P_{k+1,t} - \sum_{t=0}^{m} \lambda'_{kt} P^r_{kt} \tag{96}$$

$$P_{k+1,l} = \frac{\lambda_{kl} P_{k+1,l}}{\|\lambda_{kl} P_{k+1,l}\|} \tag{97}$$

This completes the development of the bivariate Szego polynomials as presented by Justice [Ref. 2]. The polynomial $\mathbf{P}_{kl}$ will hold the coefficients for the k x l prediction error filter. The desired causal prediction error filter, $\mathbf{A}_{kl}$, is obtained by multiplying $\mathbf{P}_{kl}$ by $z_1^{-k} z_2^{-l}$ and dividing by $p_{kl}(k,l)$ to force the leading coefficient $a_{kl}(0,0) = 1$ as was assumed. The signal generator model is found from (30) with $\mathbf{e}(n_1 n_2)$ taken as the input.

$$B(z_1, z_2) = \frac{1}{A(z_1, z_2)} = \frac{1}{1 + a_{10} z_1^{-1} + a_{01} z_2^{-1} + a_{11} z_1^{-1} z_2^{-1} + \cdots} \tag{98}$$

This again assumes that $A(z_1, z_2)$ is a causally invertible sequence.

D. LATTICE IMPLEMENTATION

1. Analysis Lattice

It is now a simple matter to utilize the orthogonal polynomials for lattice implementation. The first row is a function of z_2 only and the lattice structure is identical to the one-dimensional case. All subsequent polynomials have a connection with each of the polynomials in the previous row and each of the polynomials in the same row but of lower order in z_2.

Using the familiar form of the Levinson algorithm the first row of the lattice filter may be drawn as shown in Figure 7, given by

$$A_{0,k+1} = A_{0k} - z_2^{-1}\lambda_k A_{0k}^r \tag{99}$$

and

$$A_{0,k+1}^r = z_2^{-1}A_{0k}^r - \lambda_k A_{0k} \quad . \tag{100}$$

The orthogonal polynomials that have been developed will hold the filter coefficients in descending powers of z_1 and z_2. That is, the lowest order term in P_{kl} is the term associated with the greatest delay in the filter difference equation. We could use (94), (95) and (96) to implement a first-quadrant filter that would be statistically equivalent to the third-quadrant causal filter we desire. If all the data are stored ahead of time, which is generally the case for 2-D sequences, this would not cause any computational difficulties. We will present the causal implementation of the lattice structure. The first quadrant development is similar.

In order to correlate the powers of z_1 and z_2 with the delay and thus have a causal filter, we will define $A_{kl} = z_1^{-k}z_2^{-l}P_{kl}$. This will result in a realizable design and the form of the lattice filter. The first row of the filter is given by (99) and (100). We now assume that row k has been constructed and that we proceed to row $k+1$. Multiplying (96) by $z_1^{-(k+1)}z_2^{-l}$ yields

$$\lambda_{kl}z_1^{-(k+1)}z_2^{-l}P_{k+1,l} = z_1^{-k}z_2^{-l}P_{kl} - \sum_{t=0}^{l-1}\lambda_{kt}z_1^{-(k+1)}z_2^{-t}P_{k+1,t} - z_1^{-(k+1)}z_2^{-l}\sum_{t=0}^{m}\lambda'_{kt}P_{kt}^r \tag{101}$$

that can be written in the form

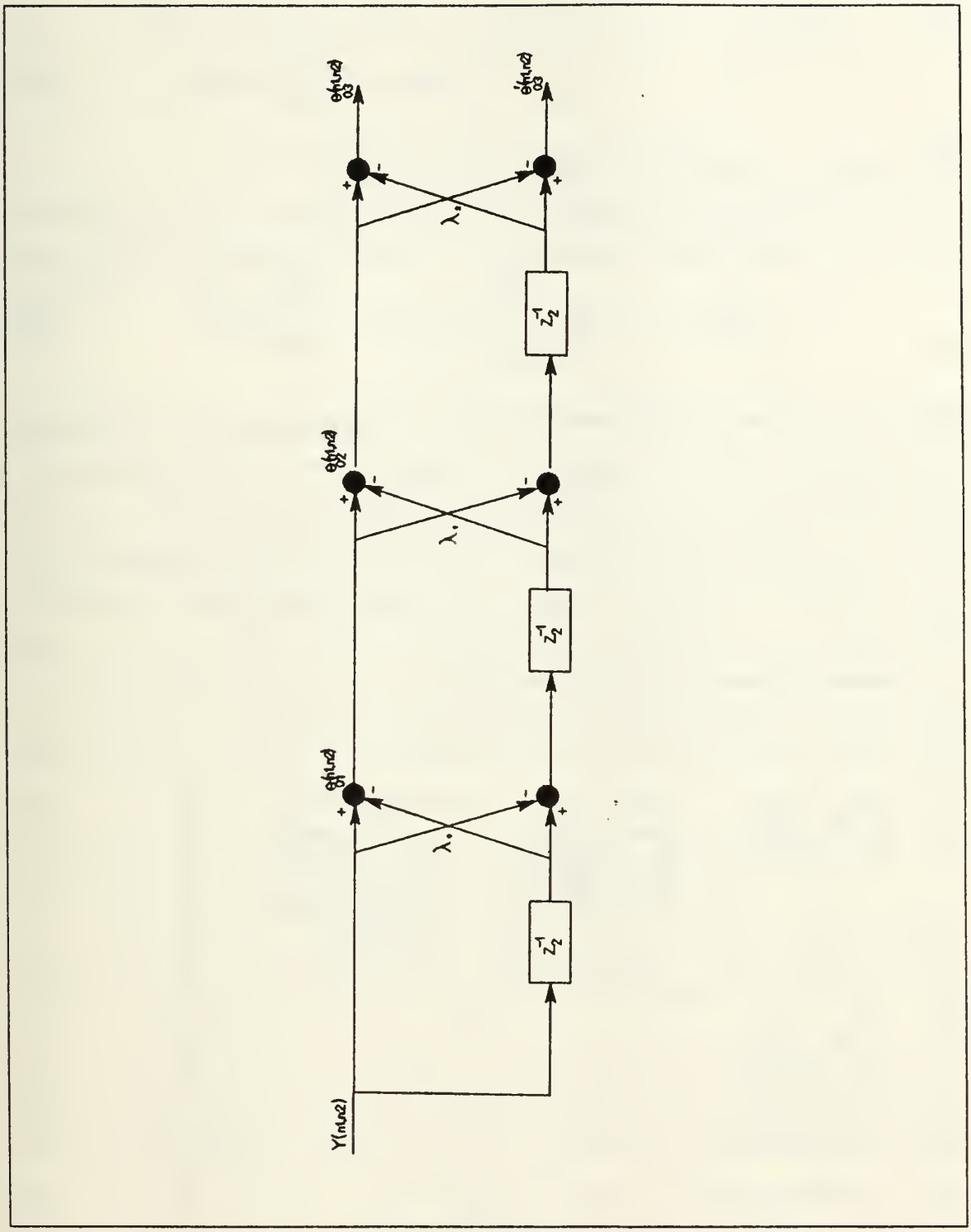


Figure 7. First row of the analysis lattice

$$\lambda_{kl}A_{k+1,l} = A_{kl} - \sum_{t=0}^{l-1} z_2^{-(l-t)}\lambda_{kt}A_{k+1,t} - z_1^{-1}\sum_{t=0}^{m}\lambda'_{kt}A^r_{kt} \quad . \tag{102}$$

Now with $A'_{k+1,l}$ defined as $z_1^{-(k+1)}z_2^{-m}A_{k+1,l}(z_1^{-1}, z_2^{-1})$, we have

$$\lambda_{kl}A^r_{k+1,l} = z_1^{-1}A^r_{kl} - \sum_{t=0}^{l-1} z_2^{(l-t)}\lambda_{kt}A^r_{k+1,t} - \sum_{t=0}^{m}\lambda'_{kt}A_{kt} \quad . \tag{103}$$

Equations (102) and (103) provide the means for implementing the two-dimensional lattice structure. Before proceeding to a schematic representation of the structure, it is worthwhile to explore the relationship of the λ_{kl}'s and the filter coefficients. This will provide some insight into the operation of the lattice filter and make it apparent that the λ's are sufficient to represent the filter. Assume that we have already found the lower order filters, A_{00}, A_{01}, A_{02}, and A_{10} and now seek A_{11}. The coefficients that are available are shown in Figure 8. By (102) we find that

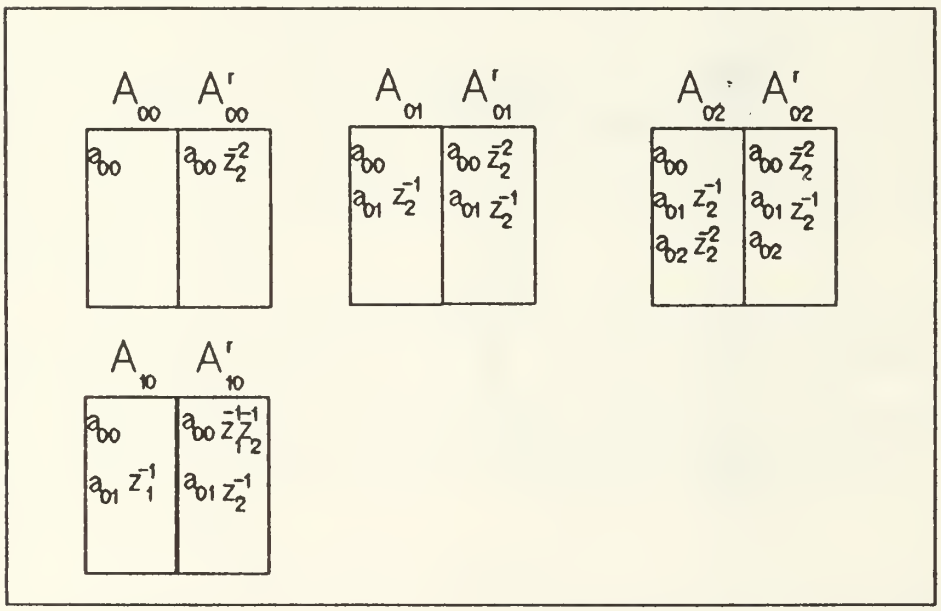


Figure 8. The lower order polynomials for A_{11}

$$\lambda_{01}A_{11} = A_{01} - z_2^{-1}\lambda_{00}A_{10} - z_1^{-1}\lambda'_{00}A'_{00} - z_1^{-1}\lambda'_{01}A'_{01} - z_1^{-1}\lambda'_{02}A'_{02} \quad . \tag{104}$$

The filter $A_{11}(z_1, z_2)$ will then be of the form

$$A_{11}(z_1, z_2) = a_{00} + a_{01}z_1^{-1} + a_{10}z_2^{-1} + a_{11}z_1^{-1}z_2^{-1} \tag{105}$$

and (104) can now be used to equate the coefficients in A_{11} with the lower order polynomials. This is shown in Figure 9. The equations for the coefficients can be written directly from Figure 9 as

$$a_{11}(00) = a_{01}(00) \tag{106}$$

$$a_{11}(01) = a_{01}(01) - \lambda_{00}a_{10}(00) \tag{107}$$

$$a_{11}(10) = -\lambda'_{02}a_{02}(02) \tag{108}$$

$$a_{11}(11) = -[\lambda_{00}a_{10}(10) + \lambda'_{01}a_{01}(01) + \lambda'_{02}a_{02}(01)] \tag{109}$$

$$0 = -[\lambda'_{00}a_{00}(00) + \lambda'_{01}a_{01}(00) + \lambda'_{02}a_{02}(00)] \quad . \tag{110}$$

It can be seen that there are enough equations to determine all the filter coefficients of A_{11}. Conversely, if the filters are known, then the λ's can be completely specified. This relationship allows the filter to be realized in either the direct form using the filter coefficients or in the lattice structure using the λ's. If the same procedure is carried out starting with (102), the coefficients in A'_{11} are found to correspond to those already found for A_{11}. This is of course a requirement if the algorithm is correct.

The second row of the lattice structure described by (102) and (103) is shown in Figure 10 for a 2x3 filter. The first row of the filter is shown in Figure 7. All subsequent rows of the filter will have the same configuration. The structure can be seen to be quite similar to a one-dimensional lattice. There are, however, two important differences. First, there is a connection to all the lower-order predictors in the row. Also the inverse delay in the reverse error path appears to be a non-causal operator. These inverse delays result from the definition of $A'_{kl} = z_1^{-k}z_2^{-m}A_{kl}(z_1^{-1}, z_2^{-1})$ and force a specific ordering of points being inputed to the lattice. For example, if we were computing the reverse error, $e'_{11}(n1,n2)$, for a filter with three horizontal filter coefficients, m = 2, then the z transforms of of these errors are given by

$$E'_{11}(z1,z2) = Y(z_1, z_2)A'_{11}(z_1,z_2) \tag{111}$$

$$E^r_{11}(z1,z2) = Y(z_1, z_2)(a_{00}z_1^{-1}z_2^{-2} + a_{01}z_1^{-1}z_2^{-1} + a_{10}z_2^{-2} + a_{11}z_2^{-1}) \quad (112)$$

$$E^r_{10}(z1,z2) = Y(z_1, z_2)(a_{00}z_1^{-1}z_2^{-2} + a_{10}z_2^{-2}) \quad . \quad (113)$$

By referring to Figure 9 we see that the coefficients $a_{11}(0,1)$ and $a_{10}(0,0)$ must be in the same power of z_1z_2 and likewise $a_{11}(1,1)$ and $a_{10}(1,0)$. Therefore, $A^r_{11}(z_1, z_2)$ must be multiplied by z_2 This would cause no problems if the direct form implementation was being used since

$$z_2E^r_{10}(z1,z2) = Y(z_1, z_2)(a_{00}z_1^{-1}z_2^{-1} + a_{10}z_2^{-1}) \quad . \quad (114)$$

There are no non-causal terms. In the lattice structure instead of linking actual filters we link error outputs. The calculation of $e^r_{11}(n1,n2)$ requires $e^r_{10}(n1,n2 + 1)$. This requires the next horizontal value of e_{10} before the present value of e_{11} can be computed. The end result of this is that the previous row errors must be completely calculated and stored before the present row errors can be computed. This forces the input to proceed across rows.

The filter shown in Figure 10 represents extending the filter order by rows. If at some point we wish to extend the filter by columns the procedure is identical and the next column would have the same connections as the row extension with the subscripts transposed.

2. Synthesis Lattice

The all-pole model provides the means to synthesize the original signal from the error sequence. The synthesis filter can also be implemented in a lattice structure. Multiplying the recursion relationships (102) and (103) by Y(z) we arrive at the recursion for the error sequences.

$$\lambda_{kl}Y(z_1, z_2)A_{k+1,l} = A_{kl}Y(z_1, z_2) - \sum_{t=0}^{l-1} z_2^{-(l-t)}\lambda_{kt}Y(z_1, z_2)A_{k+1,t} - z_1^{-1}\sum_{t=0}^{m}\lambda'_{kt}Y(z_1, z_2)A^r_{kt} \quad (115)$$

$$e_{k+1,l} = e_{kl} - \sum_{t=0}^{l-1} z_2^{-(l-t)}\lambda_{kt}e_{k+1,t} - z_1^{-1}\sum_{t=0}^{m}\lambda'_{kt}e^r_{kt} \quad (116)$$

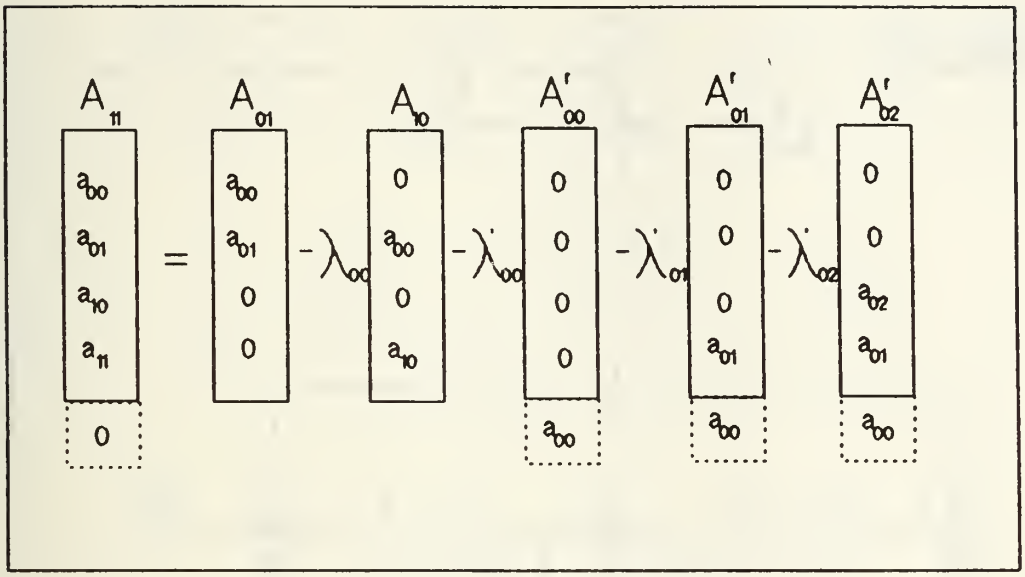


Figure 9. Coefficient relationship between orthogonal polynomials.

$$e^r_{k+1,l} = z_1^{-1} e^r_{kl} - \sum_{t=0}^{l-1} z_2^{(l-t)} \lambda_{kt} e^r_{k+1,t} - \sum_{t=0}^{m} \lambda'_{kt} e_{kt} \tag{117}$$

Since Y(n1,n2) corresponds to $\mathbf{e}_{00}$, we must write (116) in decreasing order resulting in

$$e_{k,l} = e_{k+1,l} + \sum_{t=0}^{l-1} z_2^{-(l-t)} \lambda_{kt} e_{k+1,t} + z_1^{-1} \sum_{t=0}^{m} \lambda'_{kt} e^r_{kt} \quad . \tag{118}$$

The first row of this filter is again a function of one variable and is given by

$$e_{kl} = e_{k,l+1} + \lambda_l e^r_{kl} z_2^{-1} \tag{119}$$

or

$$e^r_{k,l+1} = e^r_{kl} - \lambda_l e_{kl} \quad . \tag{120}$$

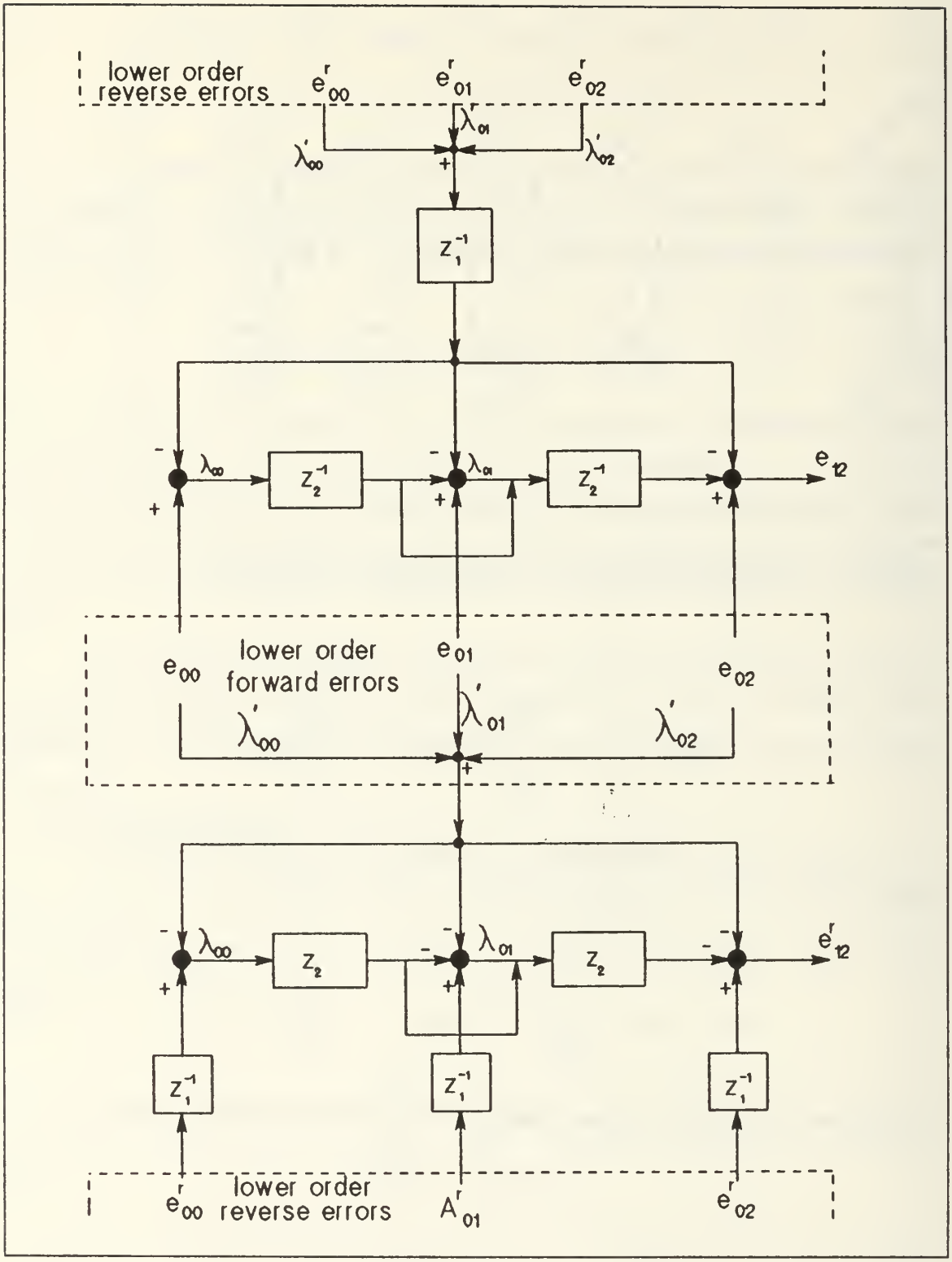


Figure 10. One cell of a 2-D analysis lattice

This is realized as shown in Figure 11, and the subsequent rows are given by (118). The signal flow graph is similar to Figure 10 with the signal flow inverted and the sign changes as indicated by (118). The node at e_{00} is shown in Figure 12. Computer simulation of these filters and the results obtained are presented in Chapter III.

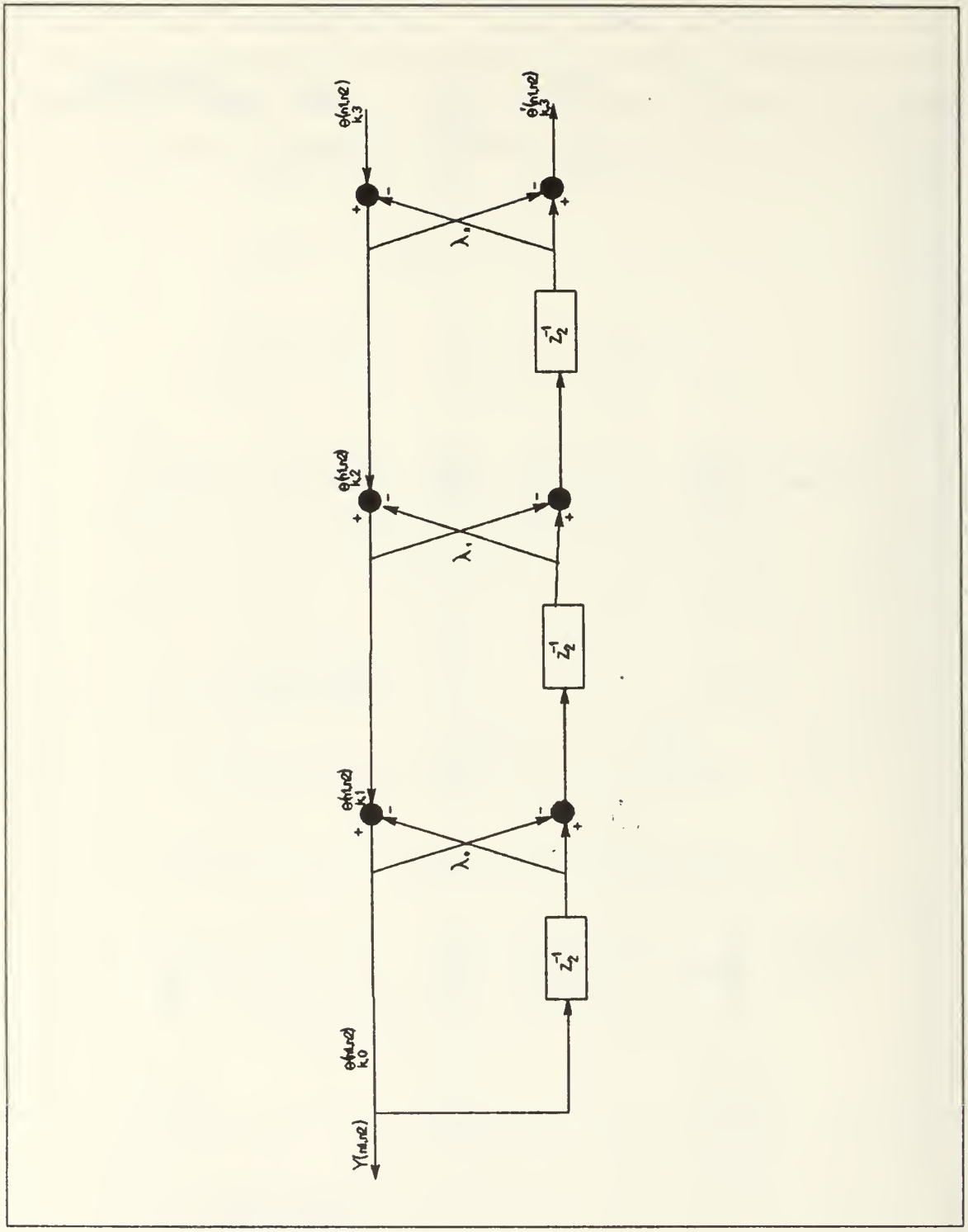


Figure 11. First row of the synthesis lattice

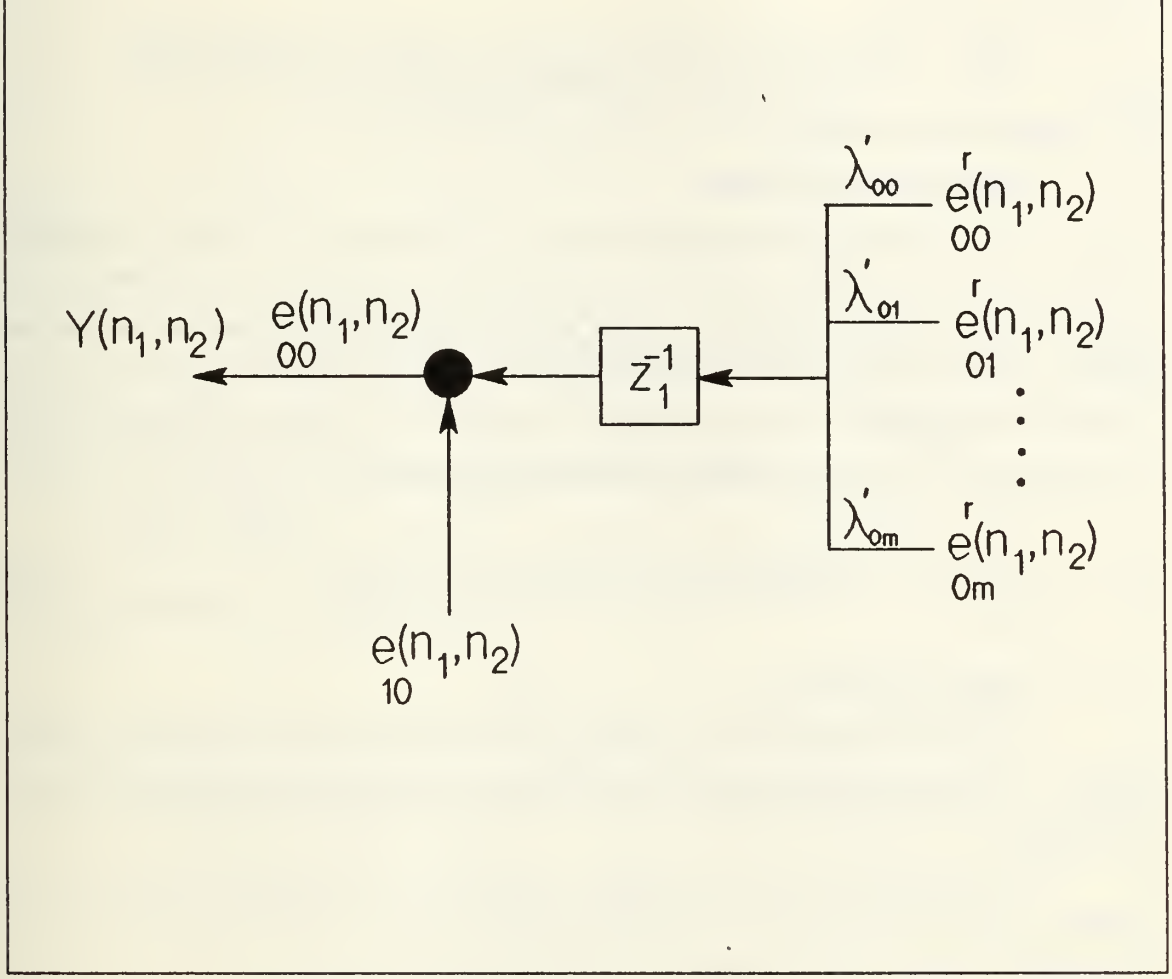


Figure 12. e_{00} node of the synthesis lattice

III. COMPUTER IMPLEMENTATION AND RESULTS

A. IMPLEMENTATION

1. Autocorrelation Function

The autocorrelation function referred to throughout this thesis is actually an estimate based on a finite sample space. There are several ways to compute the estimated autocorrelation coefficients. For a large, complex-valued sample space the most efficient method is the use of Fourier transforms. For the small real-valued data space used in the examples of this study, the autocorrelation is estimated directly from the available data. The autocorrelation coefficients are given by

$$R(n1,n2) = \sum_{k} \sum_{l} Y(k,l)\,Y(n1+k,n2+l) \quad . \tag{121}$$

2. Reflection Coefficients

The basic algorithms for calculating the coefficients were presented by Justice [Ref. 2]. With some modification these algorithms are presented in this and the following section.

The coefficients that link succesive order filters are derived in Chapter II. They will be referred to as reflection coefficients even though this term is descriptive of only the first row. The first row reflection coefficients were found to be

$$\lambda_l = \sum_{k=0}^{l} R_{0,-(k+1)} p_{0l,0k} \quad . \tag{122}$$

This result is simply the one dimensional Levinson algorithm appended with a second subscript.

For subsequent rows there are two reflection coefficients, one for the forward polynomials and one for the reverse. They are obtained as

$$\lambda_{kl} = < z_1 P_{kl},\, P_{k+1,l} > \tag{123}$$

and

$$\lambda'_{kl} = < z_1 P_{kl},\, P^r_{k,l} > \quad . \tag{124}$$

These are inner products over the autocorrelation space and are computed as

$$\lambda_{kl} = \sum_{j=t}^{l} p_{kl}(k,j)\Big[\sum_{r=0}^{k+1}\sum_{s=0}^{t} R_{k+1-r,j-s}\, p_{k+1,t}(r,s)\Big] \tag{125}$$

and

$$\lambda'_{kl} = \sum_{j=0}^{m} p_{kl}(k,m-j)\Big[\sum_{r=0}^{k}\sum_{s=0}^{l} R_{-(r+1),j-s}\, p_{k,t}(r,s)\Big] \tag{126}$$

where any negative value in a subscript results in a zero.

3. Filter Coefficients

The first row of prediction error filters are polynomials of the form P_{0k} . They can be calculated by the relationships

$$p_{0,0}(0,0) = (R_{00})^{-\frac{1}{2}} \tag{127}$$

$$p_{0,l+1}(0,j) = \left(1 - |\lambda_l|^2 p_{0,l}^2(0,l)\right)^{-\frac{1}{2}} \left(p_{0,l}(0,j-1) - \lambda_l p_{0,l}(0,l) p_{0,l}(0,l-j)\right) \tag{128}$$

$$\begin{aligned} 0 \le j \le l+1 \\ 0 \le l \le m-1 \end{aligned} \tag{129}$$

where m is the maximum filter order in the n2 direction and l is the filter order presently being computed.

All subsequent rows will increase the filter order in the n1 direction and can be recursively computed by

$$p_{k+1,l}(i,j) = p_{kl}(i-1,j) - \sum_{t=j}^{l-1} \lambda_{kl} p_{k+1,t}(i,j) - \sum_{t=m-j}^{m} \lambda'_{kl} p_{k,t}(k-i,m-j) \quad . \tag{130}$$

$$\begin{aligned} 0 \le i \le k+1 \\ 0 \le j \le l \end{aligned} \tag{131}$$

This recursion can be carried on until the desired filter order is reached. The only polynomials required to increase the filter size are the preceding row and the lower order

filters in the same row. It may be desired to normalize the filter equation, and this is easily carried out by

$$p_{k+1,l}(i,j) = \frac{p_{k+1,l}(i,j)}{d_{k+1,l}} \tag{132}$$

where

$$d_{k+1,l} = \|P_{k+1,l}\| \tag{133}$$

or

$$d_{k+1,l} = \left[p_{k+1,l}(k+1,l)\left(\sum_{f=0}^{k+1}\sum_{s=0}^{l} R_{k+1-r,l-s}\, p_{k+1,l}(r,s)\right)\right]^{\frac{1}{2}} \quad . \tag{134}$$

B. RESULTS

1. System Model

The algorithm which has been developed in this study is particularly useful in finding the autoregressive or all-pole model of the system which has generated these data. Several examples are presented in this section. Notice that the computer generated results have the negative of the desired values. This results from our original definition of the prediction of $\hat{Y}$ in (9) as

$$\hat{Y} = -\left[a_{01}\, Y(n1,n2-1) + a_{10}\, Y(n1-1,n2) \cdots\right] \quad . \tag{135}$$

Figures 13 and 14 show various size system models that were derived from the data generated by the difference equation

$$Y(n1,n2) = .4\,Y(n1-1,n2) + .6\,Y(n1,n2-1) - .6\,Y(n1-1,n2-1) + X(n1,n2)\,. \tag{136}$$

Figure 13 contains the system models resulting from driving the difference equation with an unit impulse. Figure 14 is the result of driving the system with a pseudo white noise sequence. The discrepancies in the filter coefficients can be mainly attributed to insufficient 'whiteness' in the noise data.

2. Spectral Estimation

If we assume the signal, Y(n1,n2), is generated by the model of Figure 15, the power spectrum is given by

```
FILTER ORDER--N1=2,N2=1
1.0000        -.6000
-.4000         0.6000
0.0000         0.0000


FILTER ORDER--N1=2,N2=2
1.0000        -.6000        -.0001
-.4000         0.6000        -.0000
0.0000         0.0000         0.0001


FILTER ORDER--N1=2,N2=3
1.0000        -.6000         0.0000         0.0000
-.4000         0.6000         0.0000         0.0000
0.0000         0.0000         0.0001         0.0000


FILTER ORDER--N1=3,N2=0
1.0000
-.0917
0.0000
0.0000


FILTER ORDER--N1=3,N2=1
1.0000        -.6000
-.4000         0.6000
-.0000         0.0000
0.0000        -.0000


FILTER ORDER--N1=3,N2=2
1.0000        -.6000        -.0001
-.4000         0.6000        -.0000
-.0000         0.0000         0.0001
0.0000        -.0000        -.0000


FILTER ORDER--N1=3,N2=3
1.0000        -.6000         0.0000         0.0000
-.4000         0.6000         0.0000         0.0000
0.0000         0.0000         0.0001         0.0000
0.0000        -.0000         0.0000         0.0000
```

Figure 13. System models for Eq.136 with an impulse input

```
FILTER ORDER--N1=2,N2=1
1.0000      -.5407
-.4136       0.5908
0.0107      -.0274


FILTER ORDER--N1=2,N2=2
1.0000      -.5572       0.0111
-.4125       0.6135      -.0404
0.0099      -.0327       0.0130


FILTER ORDER--N1=2,N2=3
1.0000      -.5574       0.0070       0.0307
-.4349       0.6126       0.0007      -.0287
0.0284      -.0398       -.0302       0.0699


FILTER ORDER--N1=3,N2=0
1.0000
-.2071
0.0564
-.0402


FILTER ORDER--N1=3,N2=1
1.0000      -.5402
-.4132       0.5903
0.0074      -.0216
-.0043      -.0133


FILTER ORDER--N1=3,N2=2
1.0000      -.5571       0.0139
-.4126       0.6129      -.0406
0.0073      -.0206       0.0082
-.0052      -.0188       0.0113


FILTER ORDER--N1=3,N2=3
1.0000      -.5570       0.0082       0.0302
-.4365       0.6125       0.0012      -.0285
0.0283      -.0300       -.0469       0.0781
-.0130      -.0126       0.0227      -.0190
```

Figure 14. System models for Eq.136 with a white noise sequence input

$$S(z_1, z_2) = \left(F[\,Y(n1,n2)]\right)^2 \tag{137}$$

or

$$S(z_1, z_2) = \left(F[\,B(n1,n2) * e(n1,n2)]\right)^2 \tag{138}$$

or

$$S(z_1, z_2) = (B(z_1, z_2)E(z_1, z_2))^2 \tag{139}$$

If the input is a unit variance white noise sequence, the power spectrum becomes

$$S(z_1, z_2) = B^2(z_1, z_2) \quad . \tag{140}$$

We can now use the all-pole system model to arrive at an estimate of the power spectrum

$$S(z_1, z_2) = \left(\frac{1}{A(z_1, z_2)} \right)^2 = \frac{1}{A(z_1, z_2)A^*(z_1, z_2)} \tag{141}$$

where $A(z_1, z_2)$ is the all-zero prediction error filter previously derived.

We have considered three examples of two-dimensional all-pole power spectrum estimation. In all examples the data set size used to compute the autocorrelation function is 32 x 32. In the first example a sequence consisting of two cosines at normalized frequencies (0.5,0.25) and (0.75,0.75) is corrupted by two-dimensional unit variance white noise. Figure 16 shows the surface plot of the estimated power spectrum with a 4 x 4 filter mask. The ideal response would be two impulse functions at the appropriate frequencies. The plot of Figure 16 shows several spurious peaks along a ridge which is illustrated by the contour plot in Figure 17. It can clearly be seen from the surface and contour plots in Figures 18 and 19 that the estimation of the power spectrum improves rapidly with an increased filter size. Figures 20 and 21 show the results using an 8 x 8 filter. The surface plot in Figure 20 clearly shows two discrete peaks in the power spectrum. Notice that there is a slight bias in the estimates.

In the second example three cosines at frequencies (0.5,0.25), (0.75,0.75), and (0.25,0.75) are used to generate the data. The surface and contour plots of spectral estimates are presented in Figures 22 through 27. The filter masks are square with sizes 4 x 4, 6 x 6, and 8 x 8. The ability of these filters to distiguish the spectral peaks again

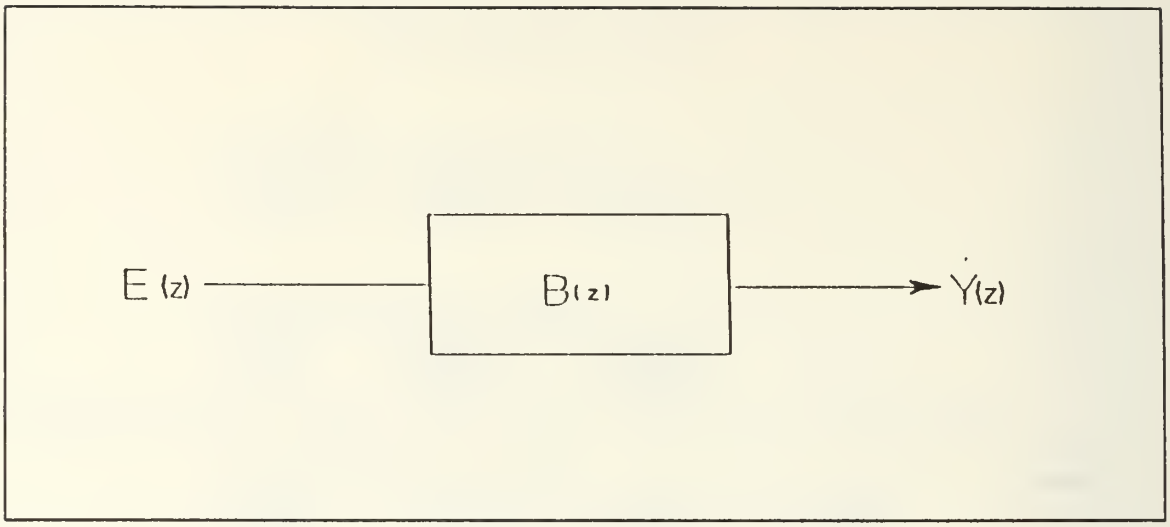


Figure 15. Generating model for spectral estimates

improves as the filter size increases from 4 x 4 to 8 x 8. While the 4 x 4 filter, as shown in Figure 22, provides very little useful information, the 8 x 8 filter in Figure 26 clearly indicates three distinct peaks.

The third example investigates the capability of the algorithm to differentiate between two closely spaced sinusoids at frequencies (0.4375,0.4375) and (0.5625,0.5625). As observed in Figures 28 through 31 the algorithm has not been able to distinguish between the two peaks. However, with a filter size 8 x 8 (see Figures 32 and 33), the spectral peaks become distinct even though there are some weak spurious peaks present.

The examples which are presented in Figures 16 to 33 show the usefulness of the algorithm in estimation of the power spectrum. The ability of the algorithm to identify discrete peaks in the power density spectrum increases rapidly with filter order. As can be seen in Figures 32 and 33, even closely spaced peaks are readily identified by an eighth-order filter. This filter requires the computation of 64 coefficients which is much simpler than using Fourier transforms. All plots are scaled from 0 to 1 which represents the interval $0,\pi$.

3. Signal Synthesis

The reflection coefficients for a single frequency data set were determined and used to implement a computer simulation of the synthesis lattice. The resulting lattice was then driven by a pseudo random white noise sequence. The spectrum of the output from the lattice is shown in Figure 34 for a 4 x 4 structure. Considering the small size of the filter and the poor quality of the input noise sequence, the results are acceptable.

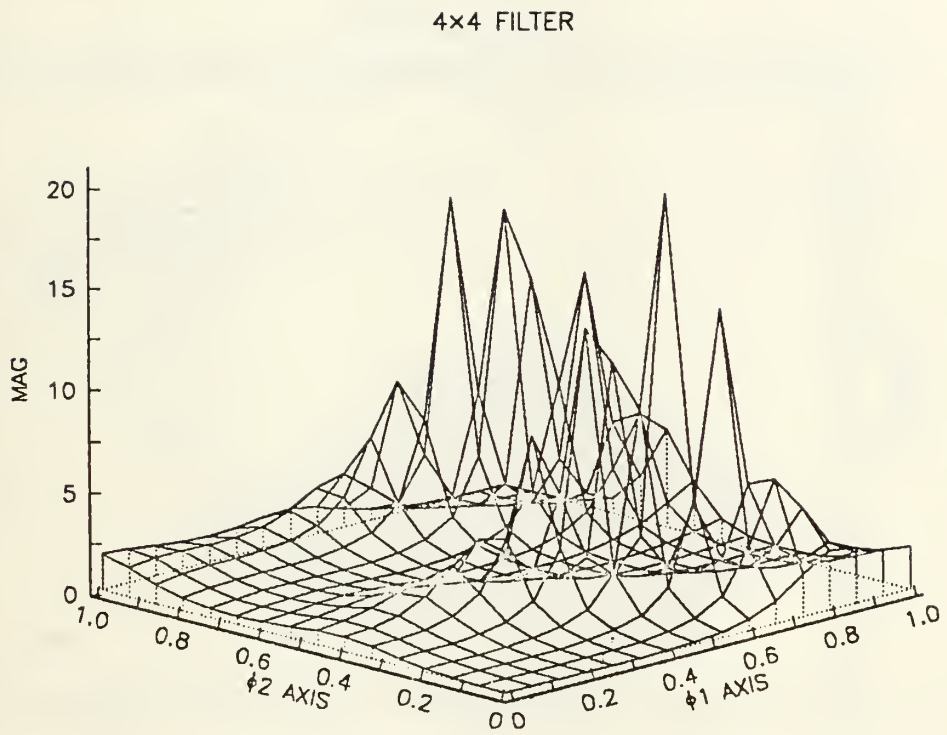


Figure 16. Surface plot of spectral estimate: $Y(n1,n2) = \cos(n1\ \pi/2 + n2\ \pi/4) + \cos(n1\ \pi\ 3/4 + n2\ \pi\ 3/4)$

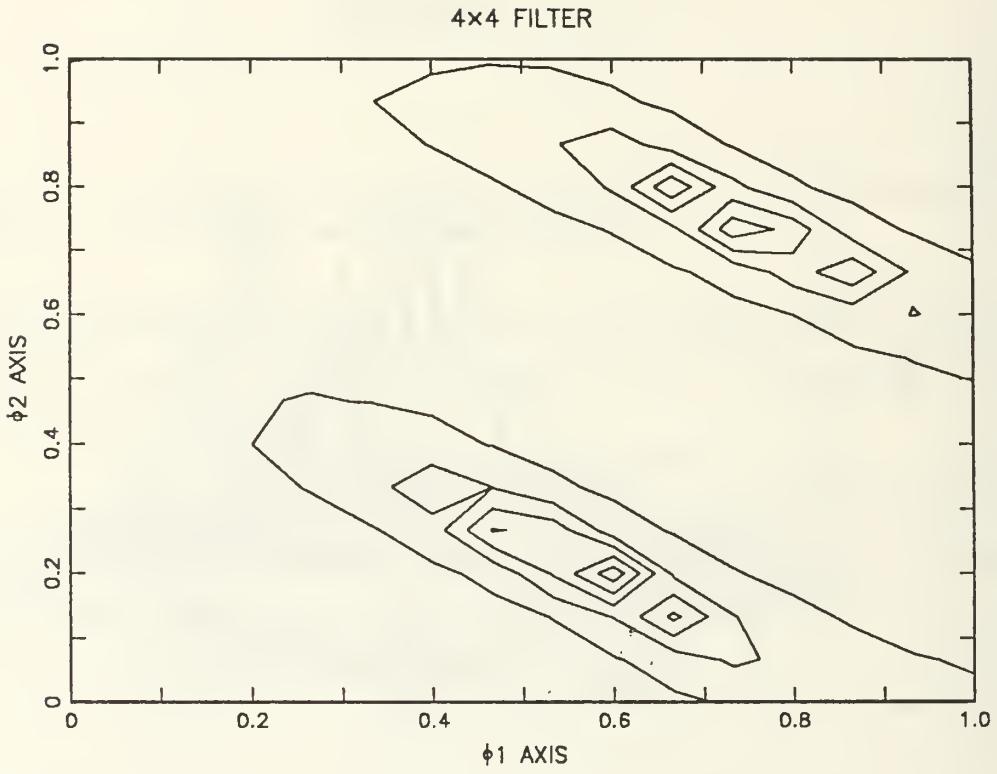


Figure 17. Contour plot of spectral estimate: $Y(n1,n2) = \cos(n1\ \pi/2 + n2\ \pi/4) + \cos(n1\ \pi\ 3/4 + n2\ \pi\ 3/4)$

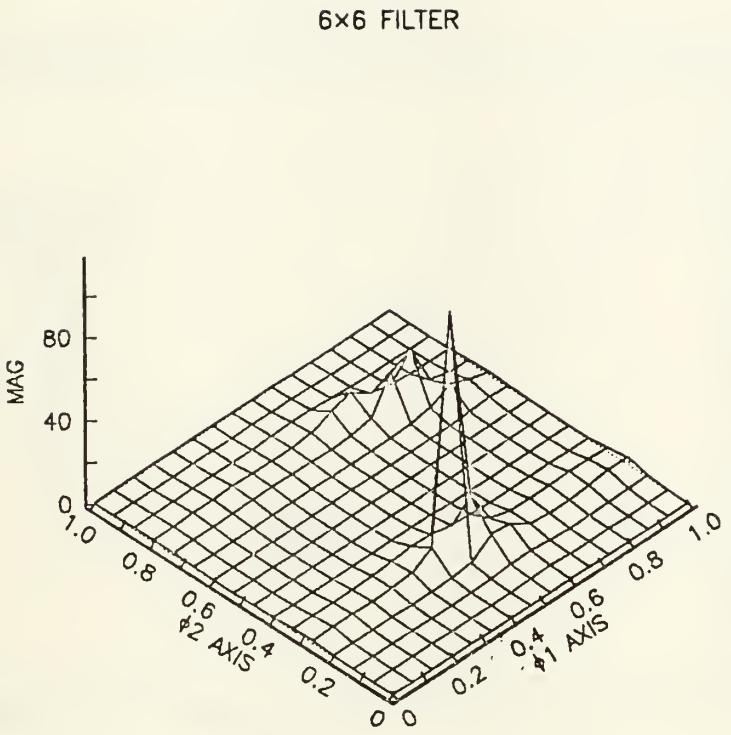


Figure 18. Surface plot of spectral estimate: $Y(n1,n2) = \cos(n1\ \pi/2 + n2\ \pi/4) + \cos(n1\ \pi\ 3/4 + n2\ \pi\ 3/4)$

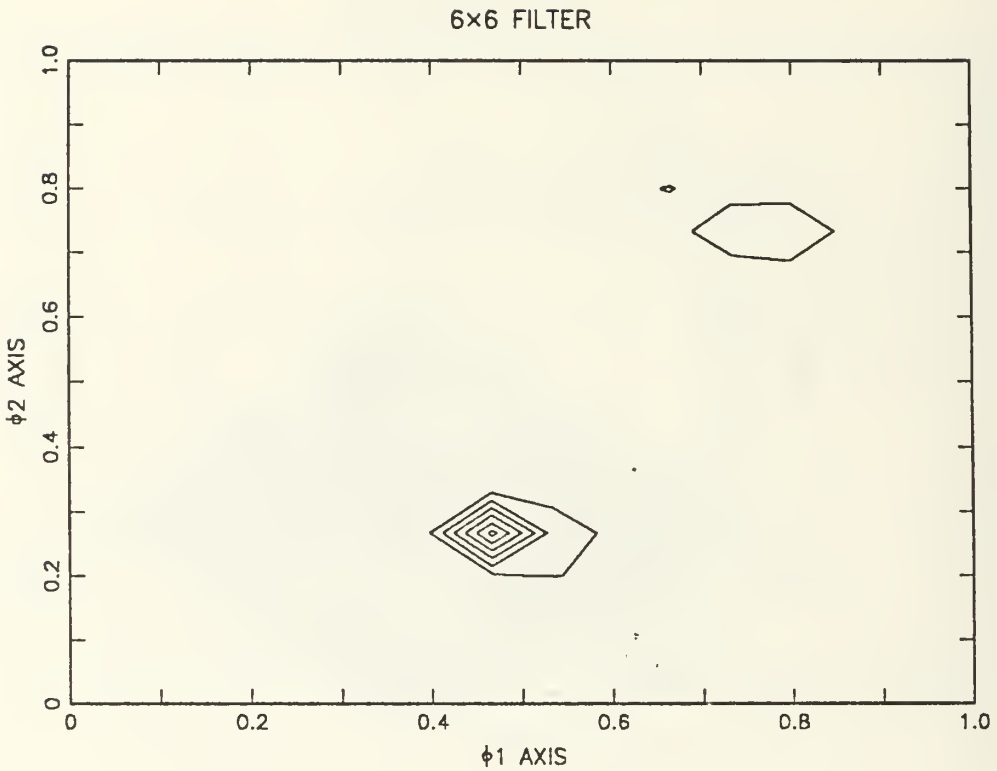


Figure 19. Contour plot of spectral estimate: $Y(n1,n2) = \cos(n1\ \pi/2 + n2\ \pi/4) + \cos(n1\ \pi\ 3/4 + n2\ \pi\ 3/4)$

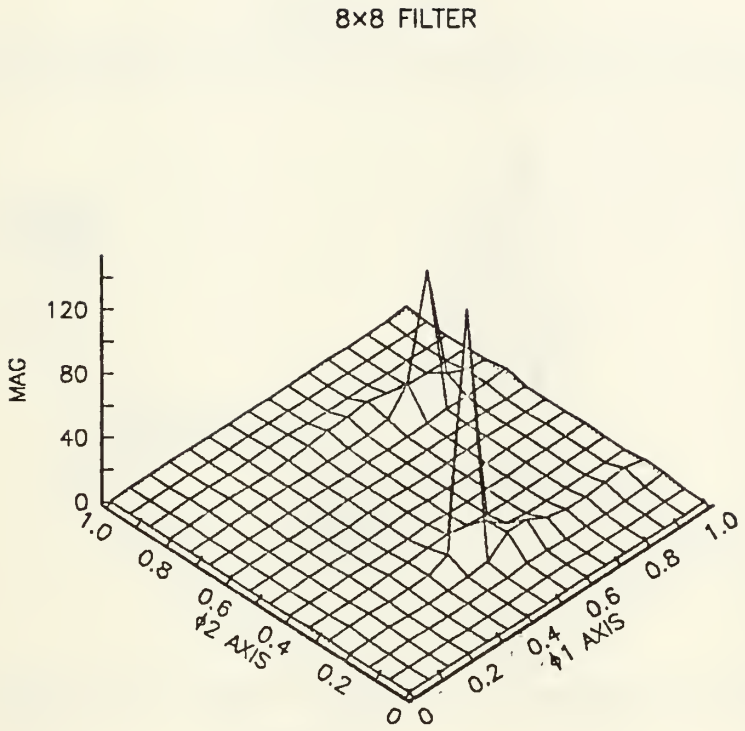


Figure 20. Surface plot of spectral estimate: $Y(n1,n2) = \cos(n1\ \pi/2 + n2\ \pi/4) + \cos(n1\ \pi\ 3/4 + n2\ \pi\ 3/4)$

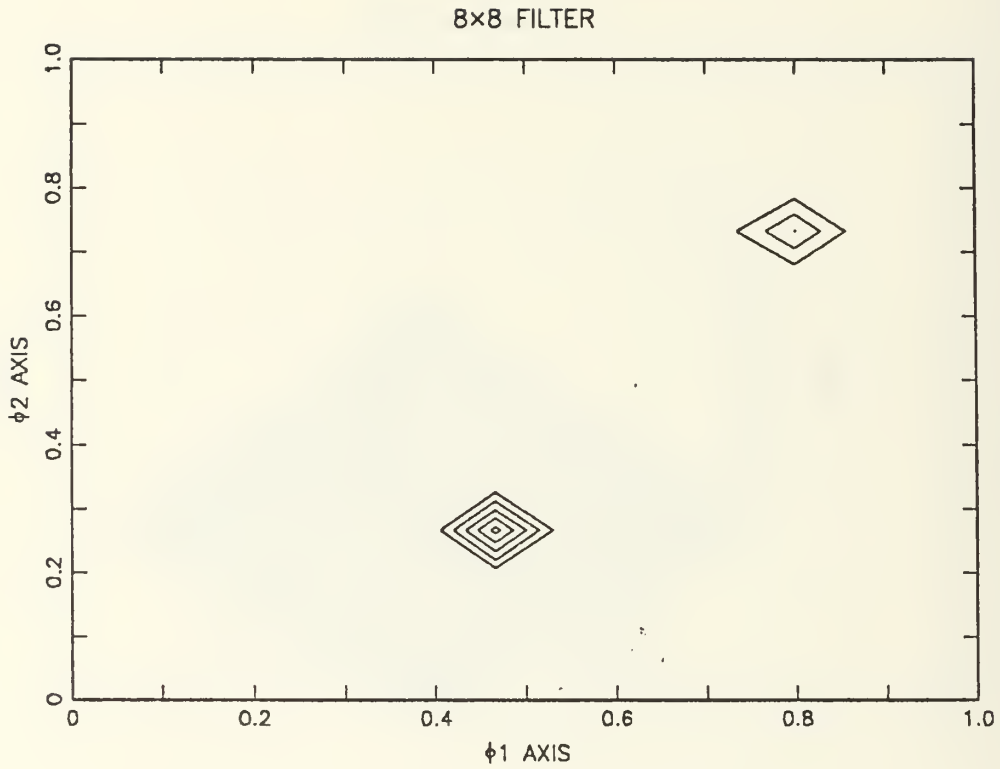


Figure 21. **Contour plot of spectral estimate:** $\Upsilon(n1,n2) = \cos(n1\ \pi/2 + n2\ \pi/4) + \cos(n1\ \pi\ 3/4 + n2\ \pi\ 3/4)$

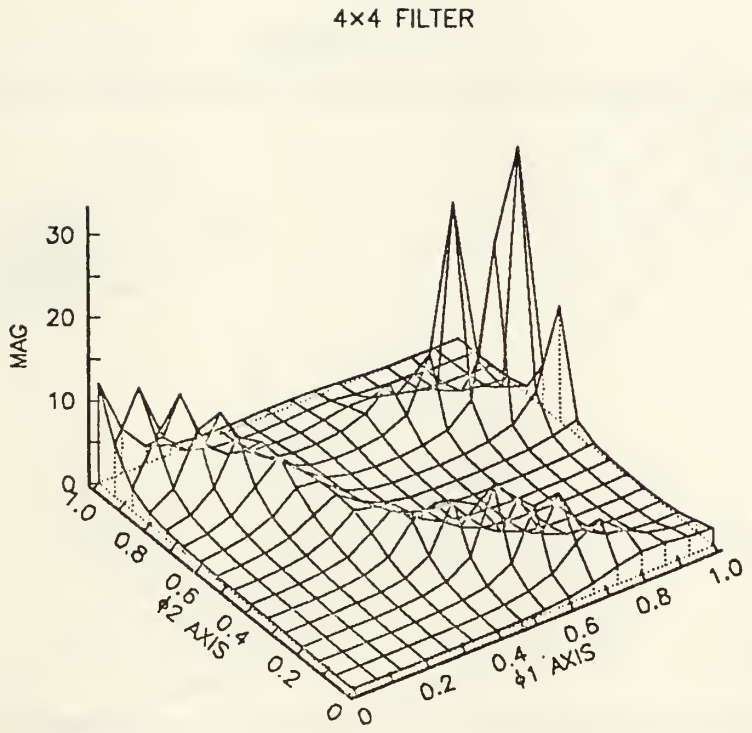


Figure 22. Surface plot of spectral estimate: $Y(n1,n2) = \cos(n1\ \pi/2 + n2\ \pi/4) + \cos(n1\ \pi\ 3/4 + n2\ \pi\ 3/4) + \cos(n1\ \pi/4, n2\ \pi\ 3/4)$

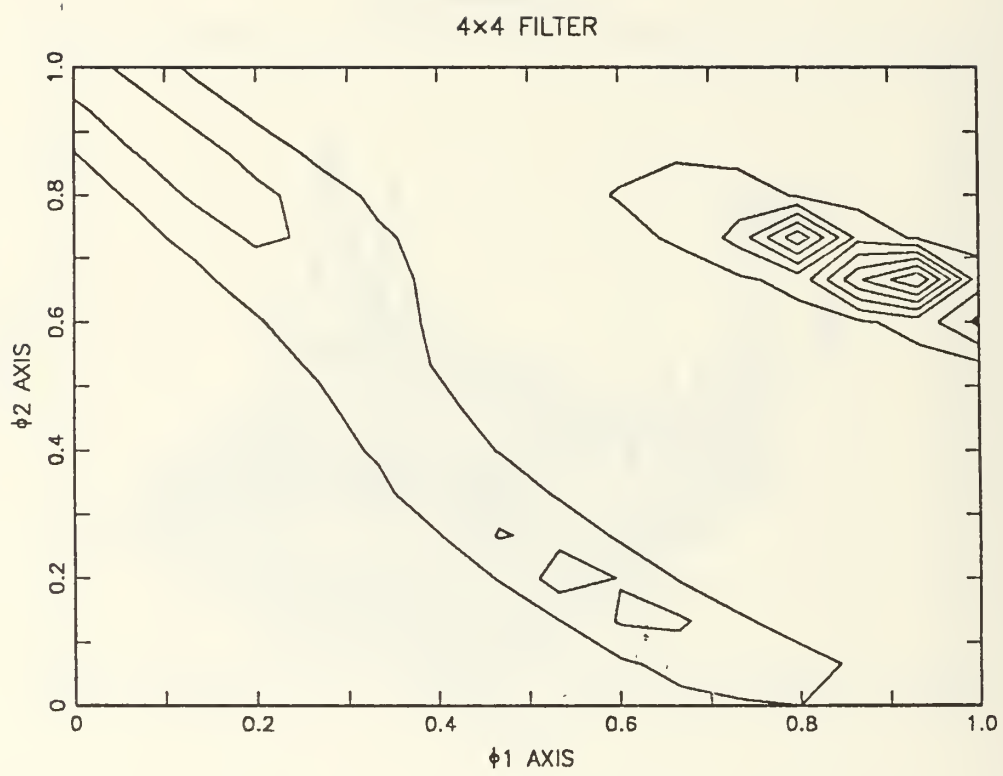


Figure 23. Contour plot of spectral estimate: $Y(n1,n2) = \cos(n1\ \pi/2 + n2\ \pi/4) + \cos(n1\ \pi\ 3/4 + n2\ \pi\ 3/4) + \cos(n1\ \pi/4, n2\ \pi\ 3/4)$

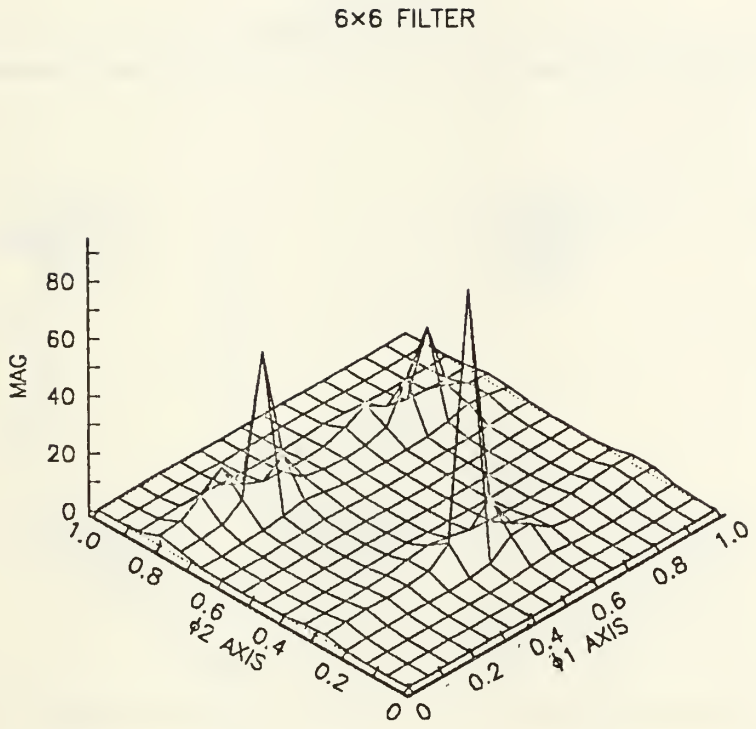


Figure 24. Surface plot of spectral estimate: $Y(n1,n2) = \cos(n1\ \pi/2 + n2\ \pi/4) + \cos(n1\ \pi\ 3/4 + n2\ \pi\ 3/4) + \cos(n1\ \pi/4, n2\ \pi\ 3/4)$

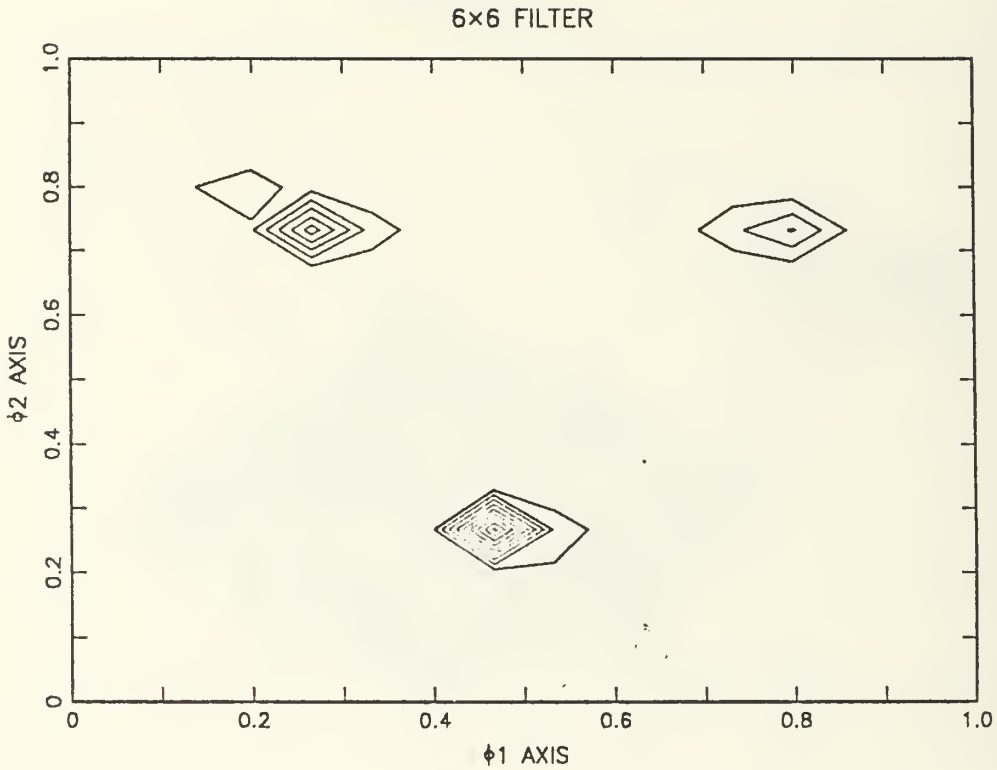


Figure 25. Contour plot of spectral estimate: $Y(n1,n2) = \cos(n1\ \pi/2 + n2\ \pi/4) + \cos(n1\ \pi\ 3/4 + n2\ \pi\ 3/4) + \cos(n1\ \pi/4, n2\ \pi\ 3/4)$

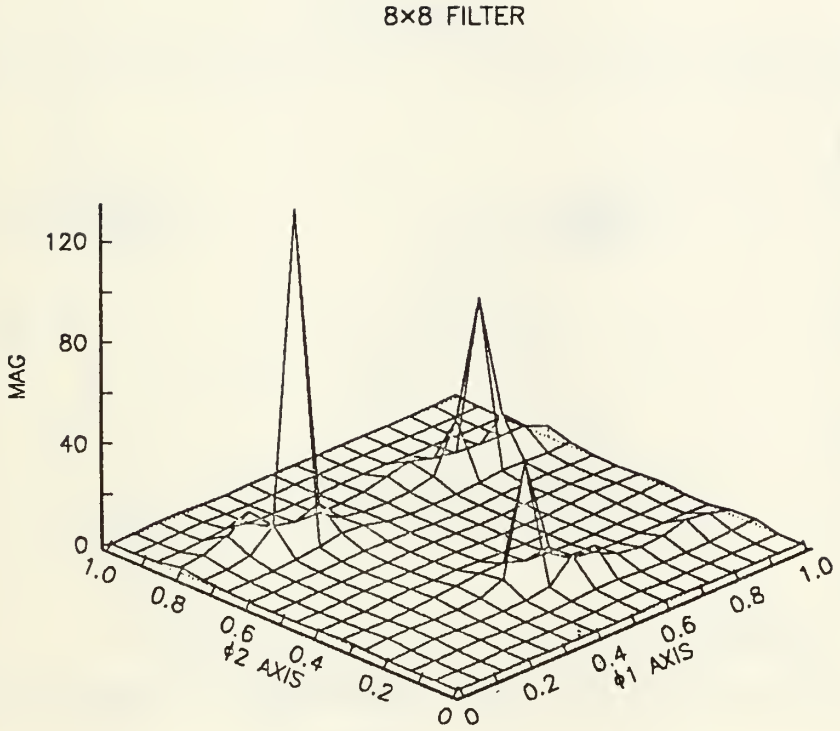


Figure 26. Surface plot of spectral estimate: $Y(n1,n2) = \cos(n1\ \pi/2 + n2\ \pi/4) + \cos(n1\ \pi\ 3/4 + n2\ \pi\ 3/4) + \cos(n1\ \pi/4, n2\ \pi\ 3/4)$

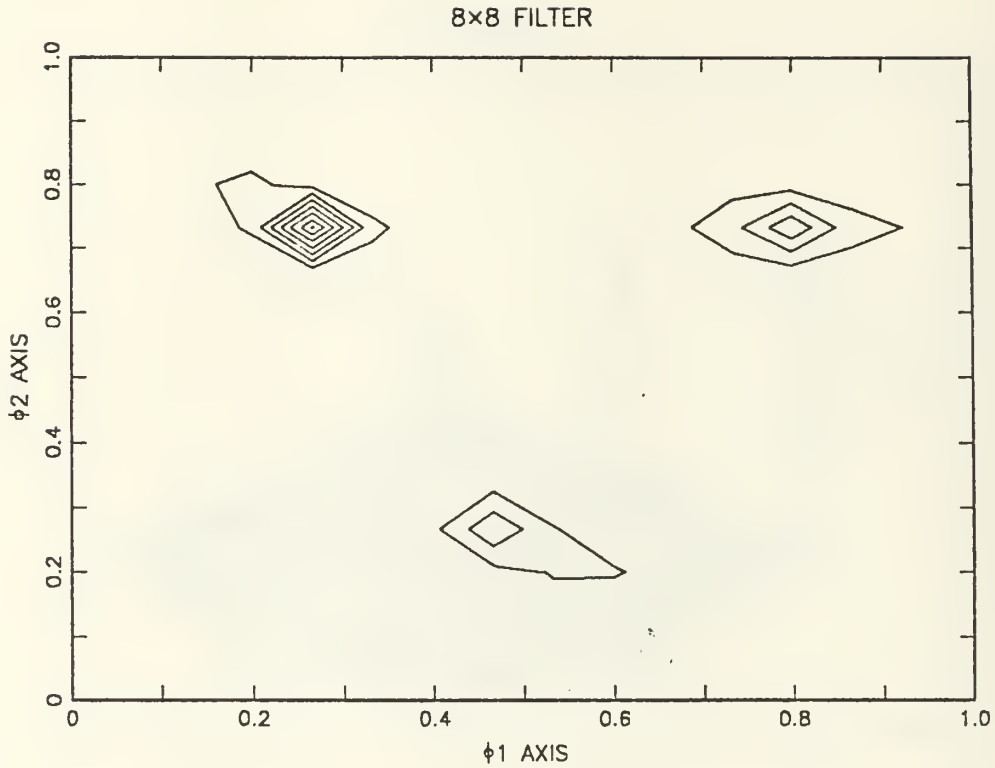


Figure 27. Contour plot of spectral estimate: $Y(n1,n2) = \cos(n1\ \pi/2 + n2\ \pi/4) + \cos(n1\ \pi\ 3/4 + n2\ \pi\ 3/4) + \cos(n1\ \pi/4, n2\ \pi\ 3/4)$

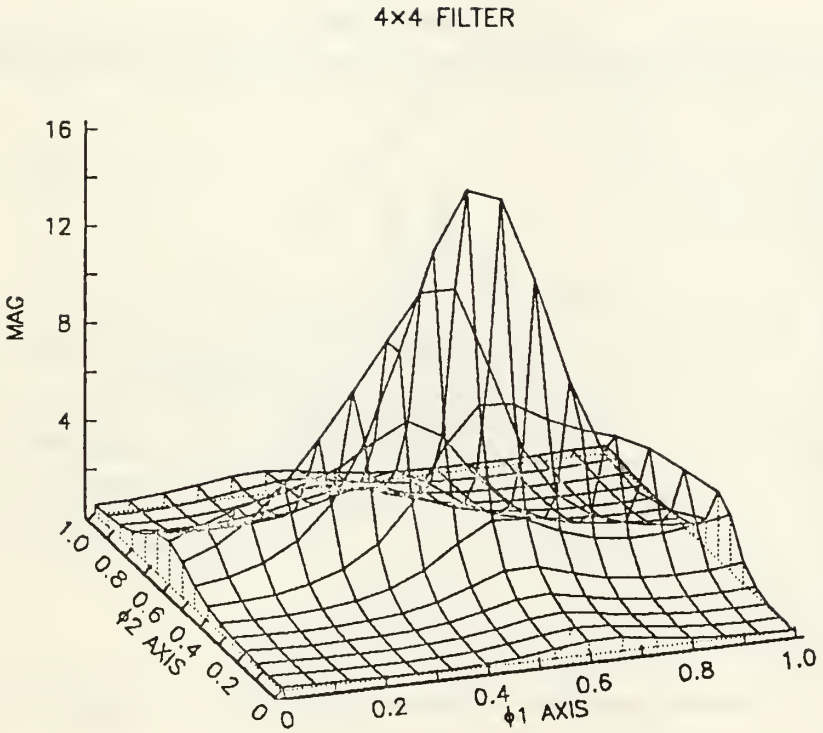


Figure 28. Surface plot of spectral estimate: $Y(n1,n2) = \cos(n1\ \pi 7/16 + n2\ \pi 7/16) + \cos(n1\pi 9/16 + n2\ \pi\ 9/16)$

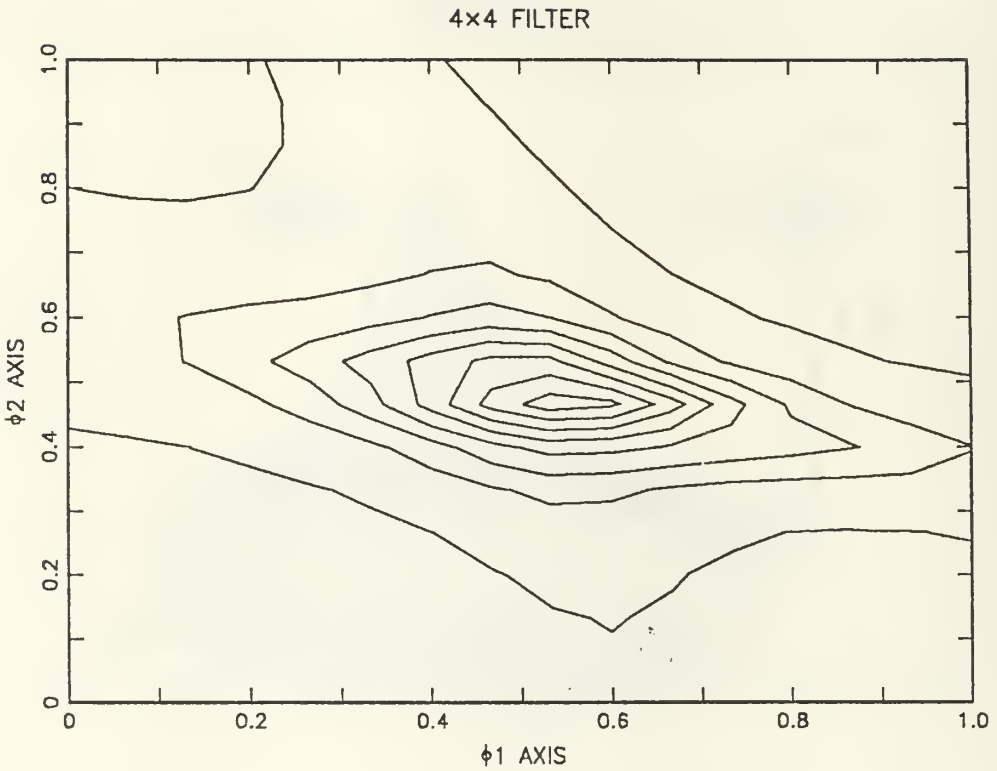


Figure 29. Contour plot of spectral estimate: $Y(n1,n2) = \cos(n1\ \pi 7/16 + n2\ \pi 7/16) + \cos(n1\pi 9/16 + n2\ \pi\ 9/16)$

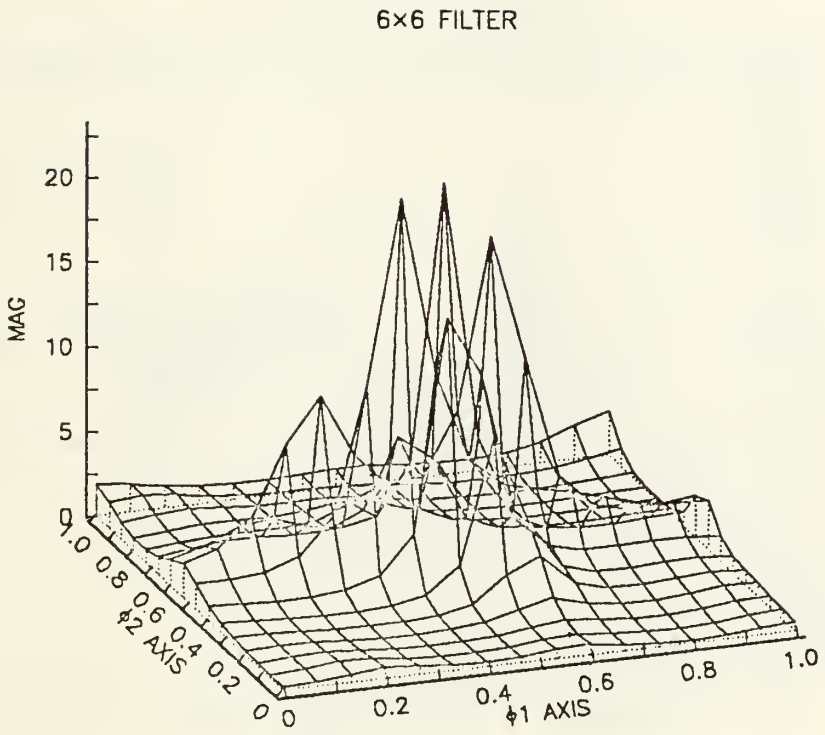


Figure 30. Surface plot of spectral estimate: $Y(n1,n2) = \cos(n1\ \pi 7/16 + n2\ \pi 7/16) + \cos(n1\pi 9/16 + n2\ \pi\ 9/16)$

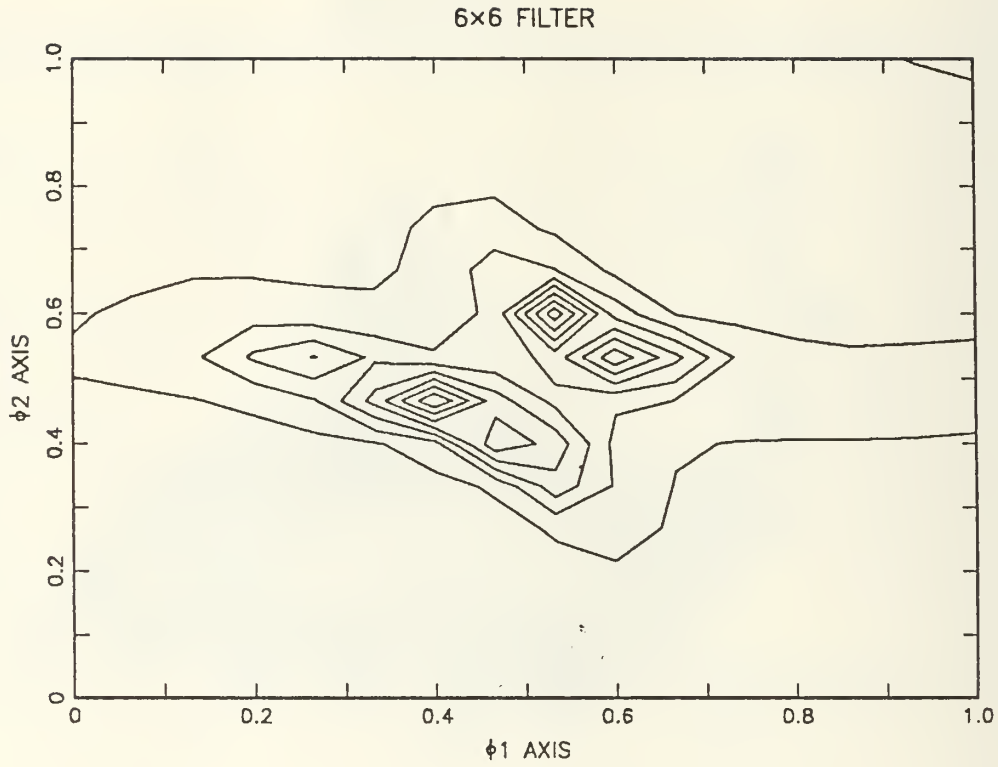


Figure 31. Contour plot of spectral estimate: $Y(n1,n2) = \cos(n1\ \pi 7/16 + n2\ \pi 7/16) + \cos(n1\pi 9/16 + n2\ \pi\ 9/16)$

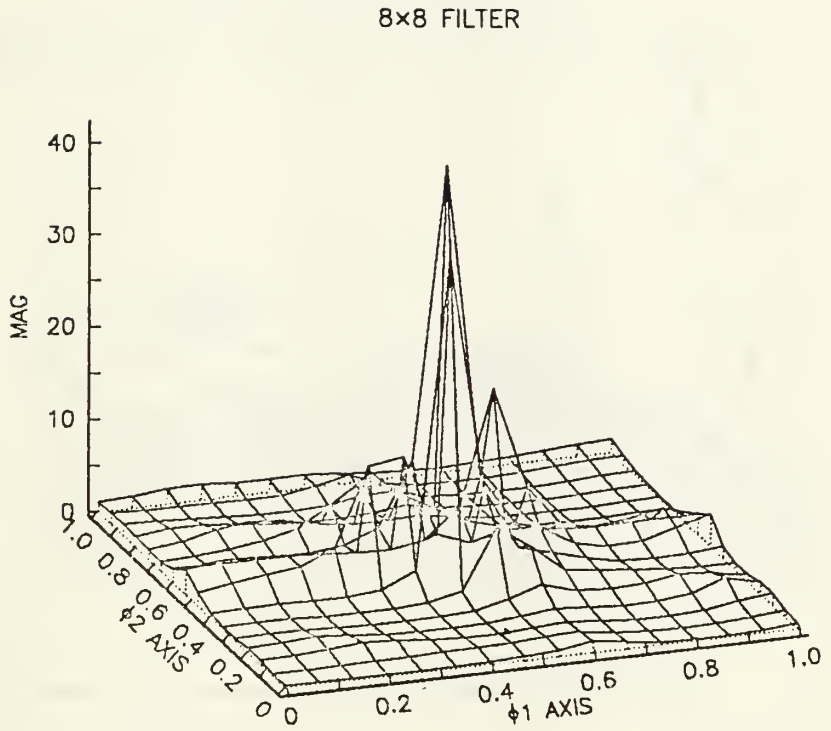


Figure 32. Surface plot of spectral estimate: $Y(n1,n2) = \cos(n1\ \pi 7/16 + n2\ \pi 7/16) + \cos(n1 \pi 9/16 + n2\ \pi\ 9/16)$

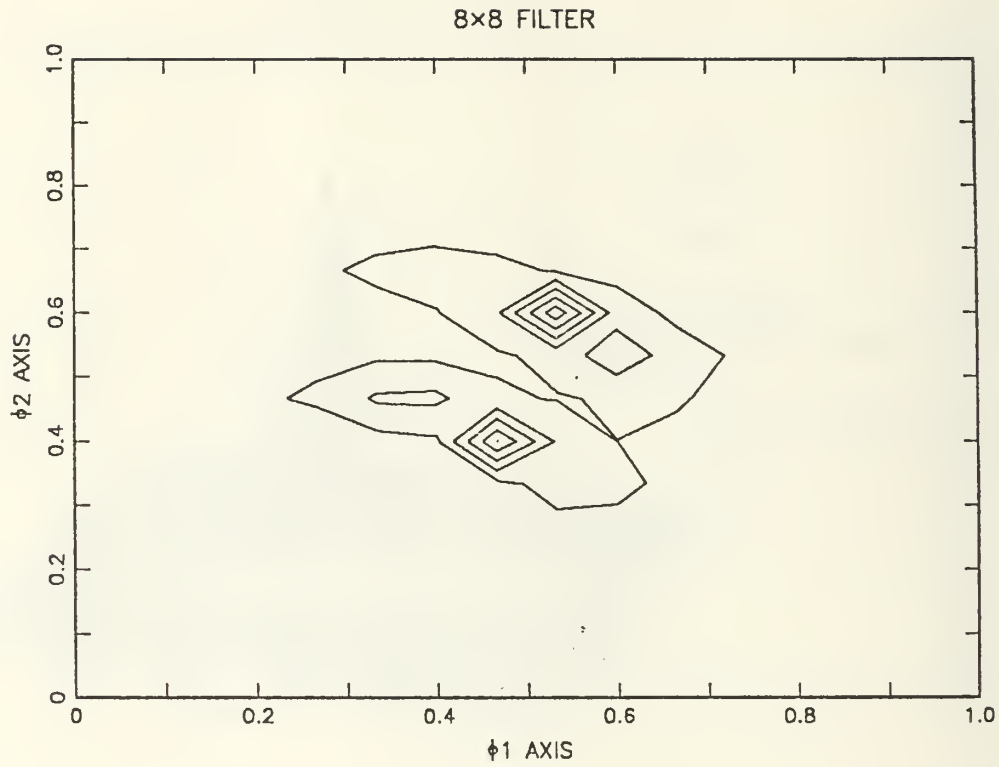


Figure 33. Contour plot of spectral estimate: $Y(n1,n2) = \cos(n1\ \pi 7/16 + n2\ \pi 7/16) + \cos(n1\pi 9/16 + n2\ \pi\ 9/16)$

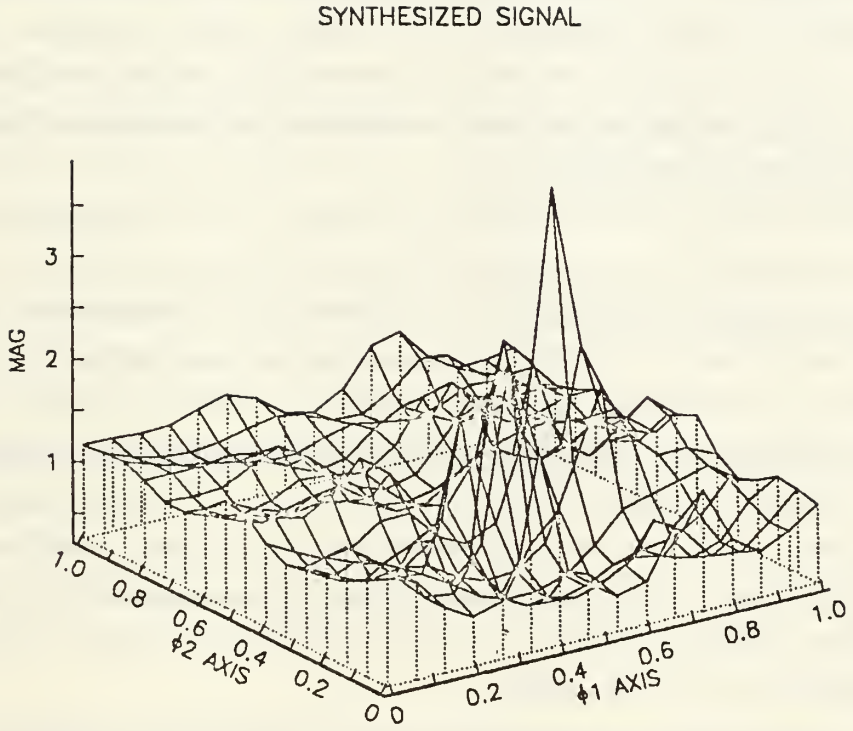


Figure 34. Spectrum of signal generated from white noise: $Y(n1,n2) = \cos(n1\ \pi/2 + n2\ \pi/4)$

IV. CONCLUSIONS

The linear prediction approach to analysis of stationary one-dimensional signals has direct applications in the study of two-dimensional signals. The solution of the 2-D normal equations in terms of orthogonal polynomials provides an efficient and effective method for system modeling and power spectral density estimation. There also exist two-dimensional lattice structures that are analogous to their one-dimensional counterparts. These lattices can be used as an alternate implementation of prediction error filters and signal syntesis filters.

The simple nature of the recursive solution to the 2-D normal equations developed in this thesis provides an excellent method for obtaining solutions to problems that are not dependent on the stability of the derived filter. For example, system modeling achieves very accurate results assuming the system being modeled is stable. Signal synthesis on the other hand can not be considered reliable since there is nothing to guarantee a stable filter. This can be attributed to the lack of autocorrelation matching in two-dimensional signals. An excellent discussion of this property can be found in [Ref. 4]. Simply stated, there are more independent autocorrelation coefficients than there are coefficients in the filters that we have found. We have ignored the second- and fourth-quadrant coefficients which are independent from the first- and second-quadrant coefficeints we have been using.

Considering the computational expense of other spectrum estimation techniques the algorithm presented in this paper is valuable if used only for that purpose. The problem of signal synthesis, while more difficult, was met with some success as evidenced by Figure 34. This limited success invites further study into autoregressive modeling of 2-D signals.

LIST OF REFERENCES

1. Marzetta, T.L., *A Linear Prediction Approach to Two-Dimensional Spectral Factorization and Spectral Estimation,* Ph.D Dissertation, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, February 1978.

2. Justice, J.H., "A Levinson-Type Algorithm for Two-Dimensional Wiener Filtering Using Bivariate Szego Polynomials," *Proc. IEEE,* v. 65, no. 6, pp. 882-86, June 1977.

3. Justice, J.H., "An Algorithm for Inverting Positive Definite Toeplitz Matrices," *SIAM J. Appl. Math.*, v. 23, no. 3, pp. 289-291.

4. Dudgeon, D.E. and Mersereau, R.M., *Multidimensional Digital Signal Processing*, Prentice-Hall, Inc., 1984.

5. Oppenheim, A.V., ed., *Applications of Digital Signal Processing* , Prentice-Hall, Inc., 1978.

6. Szego, G., *Orthogonal Polynomials*,, American Mathematical Society, 1958.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, VA 22304-6145	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, CA 93943-5002	2
3.	Chairman, Code 62 Electrical and Computer Engineering Department Naval Postgraduate School Monterey, CA 93943-5000	1
4.	Dr. Murali Tummala, Code 62 Tu Electrical and Computer Engineering Department Naval Postgraduate School Monterey, CA 93943-5000	3
5.	Professor Robert D. Strum, Code 62 St Electrical and Computer Engineering Department Naval Postgraduate School Monterey, CA 93943-5000	1
6.	Dr. Rabi N. Madan Office of Naval Research 800 North Quincy St. Arlington, VA 22217-5000	1
7.	ASWOC Sigonella ATTN: LT Greg H. Skinner FPO NY 09523-1100	2

Thesis
S5522 Skinner
c.1 Two-dimensional
auto-regressive modeling.

Thesis
S5522 Skinner
c.1 Two-dimensional
auto-regressive modeling.

AMERICAN BINDERY
TOPEKA KANSAS

thesS5522
Two-dimensional auto-regressive modeling
3 2768 000 85608 2
DUDLEY KNOX LIBRARY